

9.4

IBM MQ -Guide de configuration

IBM

Remarque

Avant d'utiliser le présent document et le produit associé, prenez connaissance des informations générales figurant à la section [«Remarques»](#), à la page 277.

Cette édition s'applique à la version 9 édition 4 d' IBM® MQ et à toutes les éditions et modifications ultérieures, sauf indication contraire dans les nouvelles éditions.

Lorsque vous envoyez des informations à IBM, vous accordez à IBM le droit non exclusif d'utiliser ou de distribuer les informations de la manière qu'il juge appropriée, sans aucune obligation de votre part.

© **Copyright International Business Machines Corporation 2007, 2024.**

Table des matières

Référence relative à la configuration.....	5
Exemple: configuration de la communication multiplateforme pour IBM MQ.....	5
Comment utiliser les exemples de communication multiplateforme.....	7
Exemple: configuration de la communication multiplateforme IBM MQ sous AIX.....	9
Exemple: configuration de la communication multiplateforme IBM MQ sous IBM i.....	16
Exemple: configuration de la communication multiplateforme IBM MQ sous Linux.....	34
Exemple: configuration de la communication multiplateforme IBM MQ sous Windows.....	41
Exemple: setting up IBM MQ cross-platform communication on z/OS.....	48
Exemple: setting up IBM MQ cross-platform communication on z/OS using QSGs.....	52
Exemple: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS.....	60
Droits d'accès au système de fichiers IBM MQ appliqués à /var/mqm.....	68
Droits d'accès aux fichiers IBM MQ dans /opt/mqm avec setuid pour mqm.....	72
Droits d'accès au système de fichiers IBM MQ sur Windows.....	74
Restrictions de dénomination pour les files d'attente.....	75
Restrictions de dénomination pour d'autres objets.....	77
Résolution de nom de file d'attente.....	79
Qu'est-ce que la résolution de nom de file d'attente?.....	80
Comment les attributs d'objet de destination sont-ils résolus pour les alias, les files d'attente éloignées et les files d'attente de cluster?.....	81
Objets système et par défaut.....	82
SYSTEM.BASE.TOPIC.....	87
Informations sur les sections des fichiers de configuration.....	88
Sections du fichier de configuration pour la mise en file d'attente répartie.....	90
Attributs de canal.....	92
Attributs de canal pour les mots clés MQSC (A-B).....	96
Attributs de canal pour les mots clés MQSC (C).....	100
Attributs de canal pour les mots clés MQSC (D-L).....	108
Attributs de canal pour les mots clés MQSC (M).....	115
Attributs de canal pour les mots clés MQSC (N-R).....	120
Attributs de canal pour les mots clés MQSC (S).....	125
Attributs de canal pour les mots clés MQSC (T-Z).....	130
Commandes et attributs de cluster IBM MQ.....	132
Attributs de cluster disponibles dans les commandes de définition de canal.....	133
Attributs de cluster disponibles dans les commandes de définition de file d'attente.....	136
Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente.....	139
DISPLAY CLUSQMgr.....	140
Actualiser le cluster.....	142
RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster.....	143
SUSPEND QMgr, RESUME QMgr et clusters.....	145
Équilibrage de charge dans les clusters.....	146
Asynchronous behavior of CLUSTER commands on z/OS.....	156
Programmes de canal.....	156
Travaux d'intercommunication sous IBM i.....	157
Etats des canaux sous IBM i.....	157
Exemple: planification d'un canal de transmission de messages sur AIX, Linux, and Windows.....	158
Configuration de l'exemple de canal de transmission de messages pour AIX, Linux, and Windows.....	159
Exécution et développement de l'exemple pour AIX, Linux, and Windows.....	161
Exemple: planification d'un canal de transmission de messages sur IBM i.....	162
Configuration de l'agent MCA sous IBM i.....	163
Exécution et développement de l'exemple pour IBM i.....	166

Example: planning a message channel on z/OS.....	167
Setting up the message channel agent on z/OS.....	168
Running and expanding the example for z/OS.....	170
Example: planning a message channel for z/OS using queue sharing groups.....	171
Setting up the queue sharing group definitions and a queue manager QM3 not in the queue sharing group.....	173
Running the queue sharing group example for z/OS.....	174
Utilisation d'un alias pour faire référence à une bibliothèque MQ.....	175
Managed File Transfer Référence de configuration.....	175
Utilisation des variables d'environnement dans les propriétés MFT.....	175
Le fichier MFT installation.properties.....	177
Le fichier MFT agent.properties.....	181
Le fichier MFT coordination.properties.....	205
Le fichier MFT command.properties.....	210
Le fichier MFT logger.properties.....	214
Sortie générée par la fonction LogTransfer.....	224
Propriétés système Java pour MFT.....	227
CipherSpecs et suites de chiffrement SHA-2 pour MFT.....	228
Fichiers de configuration du consignateur de fichiers MFT.....	228
Bibliothèque SCSQFCMD.....	236
Rubrique SYSTEM.FTE.....	237
Paramètres de file d'attente de l'agent MFT.....	239
Les files d'attente système MFT et la rubrique système.....	240
Conventions de dénomination des objets MFT.....	242
Messages d'état de l'agent MFT.....	243
IBM MQ Internet Pass-Thru Référence de configuration.....	244
Récapitulatif des propriétés MQIPT.....	245
MQIPT propriétés globales.....	251
MQIPT Propriétés de la route.....	254
mqiptAdmin propriétés.....	275
Remarques.....	277
Documentation sur l'interface de programmation.....	278
Marques.....	278

Référence relative à la configuration

Utilisez les informations de référence de cette section pour vous aider à configurer IBM MQ.

Les informations de référence de configuration sont fournies dans les sous-rubriques suivantes:

Tâches associées

Configuration


 Configuration de z/OS

Exemple: configuration de la communication multiplateforme pour IBM MQ

Cet exemple montre comment établir un réseau IBM MQ opérationnel en configurant des canaux émetteur et récepteur IBM MQ pour activer le flux de messages bidirectionnel entre les plateformes sur tous les protocoles pris en charge.










Avant de commencer

Les exemples de configuration supposent que des infrastructures réseau particulières sont en place pour des plateformes particulières:

-  z/OS communique via un contrôleur réseau 3745 (ou équivalent)

Il est également supposé que, pour SNA, toutes les définitions requises dans VTAM et le programme de contrôle de réseau (NCP) sont en place et activées pour que les plateformes connectées au réseau local puissent communiquer sur le réseau étendu (WAN). De même, pour TCP, on suppose que la fonction de serveur de noms est disponible, soit en utilisant un serveur de noms de domaine, soit en utilisant des tables locales (par exemple, un fichier hôte).

Les exemples de configuration couvrent les produits logiciels réseau suivants:

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

Pour plus d'informations sur les protocoles de communication et les logiciels pris en charge, voir [Configuration système requise pour IBM MQ](#).

Pourquoi et quand exécuter cette tâche

Cet exemple utilise les canaux émetteur et récepteur. Pour utiliser des types de canal autres que émetteur-récepteur, [DEFINE CHANNEL](#) (définir un nouveau canal).

Figure 1, à la page 6 est une représentation conceptuelle d'un canal unique et des objets IBM MQ qui lui sont associés.

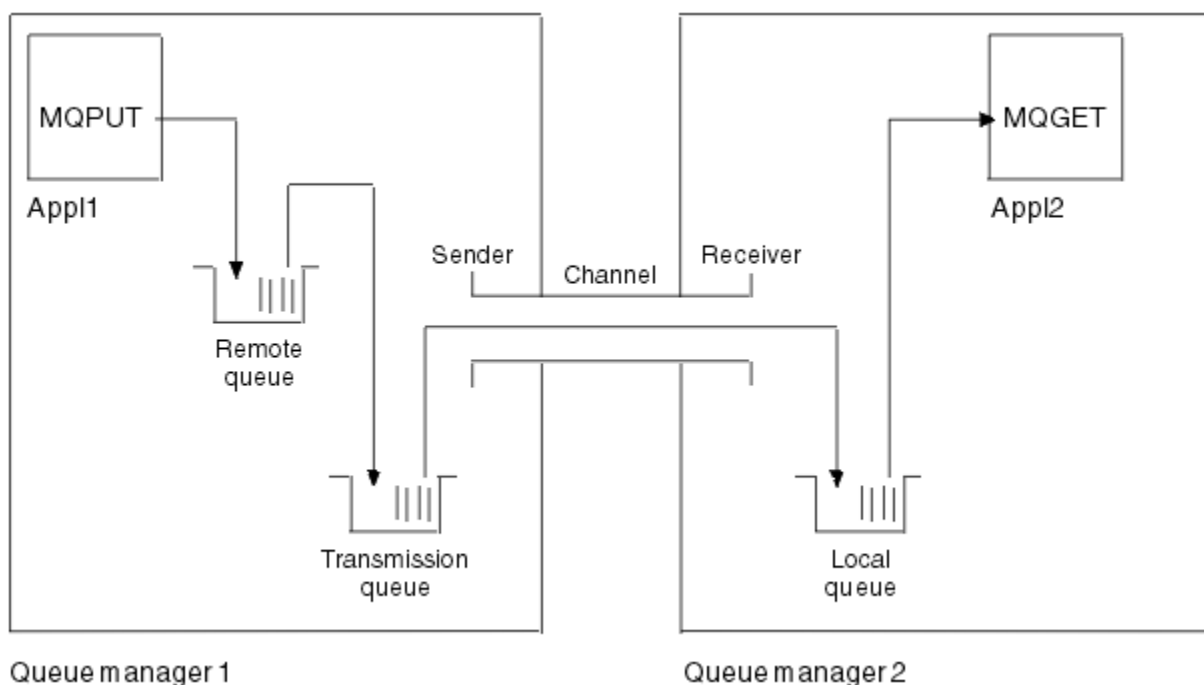


Figure 1. Canal IBM MQ à définir dans l'exemple de configuration

Cet exemple est simple, destiné à introduire uniquement les éléments de base du réseau IBM MQ . Il ne montre pas l'utilisation du déclenchement décrit dans [Déclenchement de canaux](#).

Les objets de ce réseau sont les suivants:

- Une file d'attente éloignée
- Une file d'attente de transmission
- Une file d'attente locale
- Un canal émetteur
- Un canal récepteur

App1 et App2 sont des programmes d'application ; App1 place des messages et App2 les reçoit.

App1 insère des messages dans une file d'attente éloignée. La définition de cette file d'attente éloignée indique le nom d'un gestionnaire de files d'attente cible, d'une file d'attente locale sur ce gestionnaire de files d'attente et d'une file d'attente de transmission sur ce gestionnaire de files d'attente local.

Lorsque le gestionnaire de files d'attente reçoit de App1 la demande d'insertion d'un message dans la file d'attente éloignée, il détermine, à partir de la définition de file d'attente, que la destination est éloignée. Il place donc le message, ainsi qu'un en-tête de transmission, directement dans la file d'attente de transmission spécifiée dans la définition. Le message reste dans la file d'attente de transmission jusqu'à ce que le canal soit disponible, ce qui peut se produire immédiatement.








Un canal émetteur comporte dans sa définition une référence à une, et une seule, file d'attente de transmission. Lorsqu'un canal est démarré, et à d'autres moments au cours de son fonctionnement normal, il examine cette file d'attente de transmission et envoie tous les messages qu'elle contient au système cible. Le message comporte dans son en-tête de transmission les détails de la file d'attente de destination et du gestionnaire de files d'attente.

Les exemples d'intercommunication décrivent en détail la création de chacun des objets précédents décrits, pour différentes combinaisons de plateformes.

Sur le gestionnaire de files d'attente cible, des définitions sont requises pour la file d'attente locale et le côté récepteur du canal. Ces objets fonctionnent indépendamment les uns des autres et peuvent donc être créés dans n'importe quelle séquence.

Sur le gestionnaire de files d'attente local, des définitions sont requises pour la file d'attente éloignée, la file d'attente de transmission et le côté émetteur du canal. Etant donné que la définition de file d'attente éloignée et la définition de canal font référence au nom de la file d'attente de transmission, il est conseillé de créer d'abord la file d'attente de transmission.

Procédure

1. Lisez les informations de la rubrique [«Comment utiliser les exemples de communication multiplateforme»](#), à la page 7.
2. Suivez les instructions de la ou des plateformes appropriées pour établir une connexion réseau et définir les canaux.
 - a)  Voir [«Exemple: configuration de la communication multiplateforme IBM MQ sous AIX»](#), à la page 9
 - b)  Voir [«Exemple: configuration de la communication multiplateforme IBM MQ sous IBM i»](#), à la page 16
 - c)  Voir [«Exemple: configuration de la communication multiplateforme IBM MQ sous Linux»](#), à la page 34
 - d)  Voir [«Exemple: configuration de la communication multiplateforme IBM MQ sous Windows»](#), à la page 41
 - e)  Voir [«Exemple: setting up IBM MQ cross-platform communication on z/OS»](#), à la page 48
 - f)  Voir [«Exemple: setting up IBM MQ cross-platform communication on z/OS using QSGs»](#), à la page 52
 - g)  Voir [«Exemple: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS»](#), à la page 60

Tâches associées

[Configuration de la file d'attente répartie](#)

[Configuration des communications avec d'autres gestionnaires de files d'attente sous z/OS](#)

Comment utiliser les exemples de communication multiplateforme

Les exemples de configuration pour la configuration de la communication multiplateforme pour IBM MQ décrivent les tâches effectuées sur une seule plateforme pour configurer la communication avec une autre plateforme. Les exemples décrivent ensuite les tâches permettant d'établir un canal de travail vers cette plateforme.

Dans la mesure du possible, l'objectif est de rendre l'information aussi générique que possible. Par conséquent, pour connecter deux gestionnaires de files d'attente sur des plateformes différentes, vous devez vous référer uniquement aux deux sections appropriées. Les écarts ou les cas spéciaux sont mis en évidence en tant que tels. Vous pouvez également connecter deux gestionnaires de files d'attente

s'exécutant sur la même plateforme (sur des machines différentes ou sur la même machine). Dans ce cas, toutes les informations peuvent être dérivées d'une seule section.

ALW Sous AIX, Linux, and Windows, avant de commencer à suivre les instructions pour votre plateforme, vous devez définir différentes variables d'environnement. Pour ce faire, entrez l'une des commandes suivantes:

- ▶ **Linux** ▶ **AIX** Sous AIX and Linux :

```
MQ_INSTALLATION_PATH/bin/setmqenv
```

où `MQ_INSTALLATION_PATH` correspond à l'emplacement dans lequel IBM MQ est installé. Cette commande définit les variables d'environnement pour le shell dans lequel vous travaillez actuellement. Si vous ouvrez un autre shell, vous devez entrer à nouveau la commande.

- ▶ **Windows** Sous Windows :

```
MQ_INSTALLATION_PATH/bin/setmqenv
```

où `MQ_INSTALLATION_PATH` correspond à l'emplacement dans lequel IBM MQ est installé.

Il existe des exemples dans lesquels vous pouvez trouver les paramètres utilisés dans les exemples de configuration. Il existe une brève description de chaque paramètre et des conseils sur l'emplacement des valeurs équivalentes dans votre système. Lorsque vous disposez d'un ensemble de valeurs qui vous est propre, veillez à utiliser ces valeurs lorsque vous utilisez les exemples de cette section.

Les exemples ne traitent pas de la configuration des communications dans lesquelles la mise en grappe est utilisée. Pour plus d'informations sur la configuration des communications lors de l'utilisation de la mise en cluster, voir [Configuration d'un cluster de gestionnaires de files d'attente](#). Les valeurs de configuration de communication indiquées ici s'appliquent toujours.

Il existe des exemples de configuration pour les plateformes suivantes:

- ▶ **AIX** [«Exemple: configuration de la communication multiplateforme IBM MQ sous AIX»](#), à la page 9
- ▶ **IBM i** [«Exemple: configuration de la communication multiplateforme IBM MQ sous IBM i»](#), à la page 16
- ▶ **Linux** [«Exemple: configuration de la communication multiplateforme IBM MQ sous Linux»](#), à la page 34
- ▶ **Windows** [«Exemple: configuration de la communication multiplateforme IBM MQ sous Windows»](#), à la page 41
- ▶ **z/OS** [«Exemple: setting up IBM MQ cross-platform communication on z/OS»](#), à la page 48
- ▶ **z/OS** [«Exemple: setting up IBM MQ cross-platform communication on z/OS using QSGs»](#), à la page 52
- ▶ **z/OS** [«Exemple: setting up IBM MQ cross-platform communication for intra-group queuing on z/OS»](#), à la page 60

Responsabilités informatiques

Pour comprendre la terminologie utilisée dans les exemples, prenez les instructions suivantes comme point de départ.

- Administrateur système: personne (ou groupe de personnes) qui installe et configure le logiciel pour une plateforme spécifique.

- Administrateur réseau: personne qui contrôle la connectivité du réseau local, les affectations d'adresses de réseau local, les conventions de dénomination réseau et d'autres tâches réseau. Cette personne peut appartenir à un groupe distinct ou faire partie du groupe d'administration système.

Dans la plupart des installations z/OS , un groupe est chargé de mettre à jour les logiciels ACF/VTAM, ACF/NCPet TCP/IP pour prendre en charge la configuration réseau. Les personnes de ce groupe constituent la principale source d'informations nécessaires lors de la connexion d'une plateforme IBM MQ à IBM MQ for z/OS. Ils peuvent également influencer ou imposer des conventions de dénomination de réseau sur les réseaux locaux et vous devez vérifier leur étendue de contrôle avant de créer vos définitions.

- Un type spécifique d'administrateur, par exemple l'administrateur CICS , est indiqué dans les cas où nous pouvons décrire plus clairement les responsabilités de la personne.

Les sections d'exemple de configuration ne tentent pas d'indiquer qui est responsable et capable de définir chaque paramètre. En général, plusieurs personnes différentes peuvent être impliquées.

Tâches associées

«Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5

Cet exemple montre comment établir un réseau IBM MQ opérationnel en configurant des canaux émetteur et récepteur IBM MQ pour activer le flux de messages bidirectionnel entre les plateformes sur tous les protocoles pris en charge.

Référence associée

[setmqenv](#)

Exemple: configuration de la communication multiplateforme IBM MQ sous AIX

Cet exemple montre comment configurer des liaisons de communication entre IBM MQ sur AIX et IBM MQ sur une autre plateforme et établir un canal de travail vers cette plateforme.

Avant de commencer

Pour plus d'informations sur cet exemple et sur la façon de l'utiliser, voir [«Exemple: configuration de la communication multiplateforme pour IBM MQ»](#), à la page 5 et [«Comment utiliser les exemples de communication multiplateforme»](#), à la page 7.

Pourquoi et quand exécuter cette tâche

Cet exemple décrit la configuration de la communication multiplateforme entre IBM MQ sur AIX et les plateformes suivantes:

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

Procédure

1. Etablissez une connexion réseau à l'aide de l'une des options suivantes.
 - Etablissez une connexion LU 6.2 . Pour plus d'informations sur la configuration de SNA sur TCP/IP, voir [Communications Server for AIX Library](#).
 - Etablissez une connexion TCP.

Le programme d'écoute doit être démarré explicitement avant tous les canaux. Il permet aux canaux de réception de démarrer automatiquement en réponse à une demande provenant d'un canal d'envoi entrant. Utilisez la commande suivante pour démarrer le programme d'écoute IBM MQ for TCP:

```
runmqclsr -t tcp
```

a. Editez le fichier `/etc/services`.

Remarque : Pour éditer le fichier `/etc/services`, vous devez être connecté en tant que superutilisateur ou root. Si vous ne disposez pas de la ligne suivante dans ce fichier, ajoutez-la comme suit:

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

b. Editez le fichier `/etc/inetd.conf`. Si vous ne disposez pas de la ligne suivante dans ce fichier, ajoutez-la comme indiqué, en remplaçant `MQ_INSTALLATION_PATH` par le répertoire de haut niveau dans lequel IBM MQ est installé:

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

c. Entrez la commande `refresh -s inetd`.

Remarque : Vous devez ajouter **root** au groupe `mqm`. Il n'est pas nécessaire que le groupe principal soit défini sur `mqm`. Tant que `mqm` se trouve dans l'ensemble de groupes, vous pouvez utiliser les commandes. Si vous exécutez uniquement des applications qui utilisent le gestionnaire de files d'attente, vous n'avez pas besoin des droits du groupe `mqm`.

2. Une fois la connexion établie, définissez des canaux comme décrit dans [«Configuration des canaux sur AIX»](#), à la page 10.

Configuration des canaux sur AIX

Pour configurer IBM MQ pour l'exemple de configuration sous AIX, effectuez les étapes de configuration de base pour le gestionnaire de files d'attente, puis configurez les canaux émetteur et récepteur.

Avant de commencer

Remarque :

1. Avant de commencer le processus d'installation, vérifiez que vous avez d'abord créé l'utilisateur et le groupe `mqm`, puis définissez le mot de passe.
2. Si l'installation échoue en raison d'un espace insuffisant dans le système de fichiers, vous pouvez augmenter la taille comme suit à l'aide de la commande `smitty sna`. (Utilisez `df` pour afficher l'état du système de fichiers. Indique le volume logique saturé.)

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

3. Démarrez n'importe quel canal à l'aide de la commande suivante:

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

4. Les exemples de programme sont installés dans `MQ_INSTALLATION_PATH/samp`, où `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.
5. Les journaux d'erreurs sont stockés dans `/var/mqm/qmgrs/qmgrname/errors`.

6. Sous AIX, vous pouvez démarrer une trace des composants IBM MQ à l'aide de commandes de trace IBM MQ standard ou à l'aide de la trace système AIX . Pour plus d'informations sur la trace IBM MQ et la trace système AIX , voir [Utilisation de la trace](#) .
7. Lorsque vous utilisez l'interpréteur de commandes **runmqsc** pour entrer des commandes d'administration, un signe + à la fin d'une ligne indique que la ligne suivante est une continuation. Assurez-vous qu'il existe un espace entre le dernier paramètre et le caractère de continuation.

Procédure

1. Pour effectuer la configuration de base du gestionnaire de files d'attente, procédez comme suit:
 - a) Créez le gestionnaire de files d'attente à partir de la ligne de commande AIX à l'aide de la commande suivante:

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

où :

aix

Est le nom du gestionnaire de files d'attente

-q

Indique qu'il doit devenir le gestionnaire de files d'attente par défaut

-u *dlqname*

Indique le nom de la file d'attente de messages non distribuables

Cette commande crée un gestionnaire de files d'attente et un ensemble d'objets par défaut.

- b) Démarrez le gestionnaire de files d'attente à partir de la ligne de commande AIX à l'aide de la commande suivante:

```
strmqm aix
```

où *aix* est le nom donné au gestionnaire de files d'attente lors de sa création.

- c) Démarrez **runmqsc** à partir de la ligne de commande AIX et utilisez-la pour créer la file d'attente de messages non distribuables en entrant la commande suivante:

```
def ql (dlqname)
```

où *dlqname* est le nom donné à la file d'attente de messages non distribuables lors de la création du gestionnaire de files d'attente.

2. Configurez les canaux pour l'exemple de configuration.

Pour plus d'informations sur les paramètres utilisés dans les exemples suivants, voir [«Paramètres de configuration de canal pour AIX»](#), à la page 13. Dans chaque cas, l'exemple montre la commande MQSC. Démarrez **runmqsc** à partir d'une ligne de commande AIX et entrez chaque commande à tour de rôle, ou générez les commandes dans un fichier de commandes.

Windows Ces exemples permettent de connecter IBM MQ sous AIX à IBM MQ sous Windows. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs appropriées des tables de [«Paramètres de configuration de canal pour AIX»](#), à la page 13 à la place des valeurs de Windows.

- a) Définissez le canal émetteur comme illustré dans les exemples suivants:

- Utilisation de SNA:

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

def qr (WINNT.REMOTEQ) +                       D
  rname(WINNT.LOCALQ) +                       E
  rqnname(WINNT) +                             C
```

```

xmitq(WINNT) +
replace
F

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

```

- Utilisation de TCP:

```

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

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

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

```

b) Définissez le canal récepteur comme illustré dans les exemples suivants:

- Utilisation de SNA:

```

def ql (AIX.LOCALQ) replace
B

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

```

- Utilisation de TCP:

```

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

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

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

```

Remarque : Il existe d'autres moyens de s'assurer que les canaux récepteurs SNA s'activent correctement lorsqu'un canal émetteur lance une conversation.

Lors du processus de configuration de AIX Communications Server, un profil TPN LU 6.2 a été créé, qui contenait le chemin d'accès complet à un programme exécutable TP. Dans l'exemple, le fichier s'appelait u/interops/AIX.crs6a. Vous pouvez choisir un nom, mais envisagez d'y inclure le nom de votre gestionnaire de files d'attente. Le contenu du fichier exécutable doit être:

```

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

```

où *aix* est le nom du gestionnaire de files d'attente (A) et *MQ_INSTALLATION_PATH* est le répertoire de haut niveau dans lequel IBM MQ est installé. Après avoir créé ce fichier, activez-le pour exécution en exécutant la commande suivante:

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

Comme alternative à la création d'un fichier exécutable, vous pouvez spécifier le chemin d'accès dans le panneau Add LU 6.2 TPN Profile, à l'aide des paramètres de ligne de commande.

La spécification d'un chemin de l'une de ces deux manières permet de s'assurer que les canaux récepteurs SNA s'activent correctement lorsqu'un canal émetteur lance une conversation.

AIX Paramètres de configuration de canal pour AIX

Paramètres requis pour configurer les canaux pour l'exemple de configuration sur AIX.

L'étape «2», à la page 11 de «Configuration des canaux sur AIX», à la page 10 décrit la configuration à effectuer sur le gestionnaire de files d'attente AIX pour implémenter le canal décrit dans «Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5. Les exemples dans «Configuration des canaux sur AIX», à la page 10 permettent de connecter IBM MQ for IBM i et IBM MQ for Windows. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs de la table appropriée à la place des valeurs de Windows.

Remarque : Les mots en **gras** sont des valeurs suggérées et reflètent les noms des objets IBM MQ utilisés dans ces exemples. Vous pouvez les modifier dans l'installation de votre produit mais, si vous le faites, veillez à utiliser vos propres valeurs lorsque vous utilisez les exemples de cette section.

Définition du noeud local

Tableau 1. Exemples de configuration pour la définition du noeud local

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
A	Nom du gest. de files		AIX
B	Nom de la file d'attente locale		AIX.LOCALQ

Connexion à IBM MQ sur Windows

Windows

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour Windows», à la page 46, comme indiqué.

Tableau 2. Exemples de configuration pour la connexion à IBM MQ sous Windows

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	winnt
D	Nom de la file d'attente distante		WINNT.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	WINNT.LOCALQ
F	Nom de la file d'attente de transmission		winnt
G	Nom du canal émetteur (SNA)		AIX.WINNT.SNA
H	Nom du canal émetteur (TCP/IP)		AIX.WINNT.TCP

Tableau 2. Exemples de configuration pour la connexion à IBM MQ sous Windows (suite)

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
I	Nom de canal récepteur (SNA)	G	WINNT.AIX.SNA
J	Nom du canal récepteur (TCP)	H	WINNT.AIX.TCP

Connexion à IBM MQ sur Linux

Linux

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour Linux», à la page 39, comme indiqué.

Tableau 3. Exemples de configuration pour la connexion à IBM MQ sous Linux

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	Linux
D	Nom de la file d'attente distante		LINUX.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	LINUX.LOCALQ
F	Nom de la file d'attente de transmission		Linux
G	Nom du canal émetteur (SNA)		AIX.LINUX.SNA
H	Nom du canal émetteur (TCP/IP)		AIX.LINUX.TCP
I	Nom de canal récepteur (SNA)	G	LINUX.AIX.SNA
J	Nom du canal récepteur (TCP/IP)	H	LINUX.AIX.TCP

Connexion à IBM MQ sur IBM i

IBM i

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour IBM i», à la page 31, comme indiqué.

Tableau 4. Exemples de configuration pour la connexion à IBM MQ sous IBM i

division d'identification	Parameter Name	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	AS400
D	Nom de la file d'attente distante		AS400.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	AS400.LOCALQ
F	Nom de la file d'attente de transmission		AS400
G	Nom du canal émetteur (SNA)		AIX.AS400.SNA
H	Nom du canal émetteur (TCP)		AIX.AS400.TCP

Tableau 4. Exemples de configuration pour la connexion à IBM MQ sous IBM i (suite)

division d'identification	Parameter Name	Référence	Exemple utilisé
I	Nom de canal récepteur (SNA)	G	AS400.AIX.SNA
J	Nom du canal récepteur (TCP)	H	AS400.AIX.TCP

connexion à IBM MQ for z/OS



Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Channel configuration parameters for z/OS», à la page 50, comme indiqué.

Tableau 5. Exemples de configuration pour la connexion à IBM MQ for z/OS

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	 système MVS
D	Nom de la file d'attente distante		MVS.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	MVS.LOCALQ
F	Nom de la file d'attente de transmission		 système MVS
G	Nom du canal émetteur (SNA)		AIX.MVS.SNA
H	Nom du canal émetteur (TCP)		AIX.MVS.TCP
I	Nom de canal récepteur (SNA)	G	MVS.AIX.SNA
J	Nom du canal récepteur (TCP)	H	MVS.AIX.TCP

Connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente



Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Shared channel configuration parameters», à la page 59, comme indiqué.

Tableau 6. Exemples de configuration pour la connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	QSG
D	Nom de la file d'attente distante		QSG.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	QSG.SHAREDQ
F	Nom de la file d'attente de transmission		QSG
G	Nom du canal émetteur (SNA)		AIX.QSG.SNA
H	Nom du canal émetteur (TCP)		AIX.QSG.TCP

Tableau 6. Exemples de configuration pour la connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente (suite)

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
I	Nom de canal récepteur (SNA)	G	QSG.AIX.SNA
J	Nom du canal récepteur (TCP)	H	QSG.AIX.TCP

IBM i Exemple: configuration de la communication multiplateforme IBM MQ sous IBM i

Cet exemple montre comment configurer des liaisons de communication entre IBM MQ sur IBM i et IBM MQ sur une autre plateforme et établir un canal de travail vers cette plateforme.

Avant de commencer

Pour plus d'informations sur cet exemple et sur la façon de l'utiliser, voir «Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5 et «Comment utiliser les exemples de communication multiplateforme», à la page 7.

Pourquoi et quand exécuter cette tâche

Cet exemple décrit la configuration de la communication multiplateforme entre IBM MQ sur IBM i et les plateformes suivantes:

-  Windows
-  AIX
-  Linux
-  z/OS ou MVS
- VSE/ESA

Procédure

1. Etablissez une connexion réseau à l'aide de l'une des options suivantes.
 - Etablissez une connexion LU 6.2 comme décrit dans «Etablissement d'une connexion LU 6.2 sous IBM i», à la page 16.
 - Etablissez une connexion TCP comme décrit dans «Etablissement d'une connexion TCP sous IBM i», à la page 26.
2. Une fois la connexion établie, définissez des canaux comme décrit dans «Configuration des canaux sur IBM i», à la page 28.

IBM i Etablissement d'une connexion LU 6.2 sous IBM i

Pour établir une connexion LU 6.2 sous IBM i, vous devez configurer le noeud local et le connecter à un noeud partenaire.

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur les paramètres nécessaires à la configuration de la communication entre le système IBM i et l'une des autres plateformes IBM MQ, voir les tableaux dans «Paramètres de

configuration d'une connexion LU 6.2 sous IBM i», à la page 21. Les nombres entre crochets () dans les étapes de la tâche correspondent aux valeurs de la colonne *ID* de ces tables.

Pour configurer le noeud local, vous devez:

- Créer une description de ligne
- Ajoutez un poste de routage, puis démarrez le sous-système

Pour vous connecter à un noeud partenaire, vous devez:

- Création d'une description de contrôleur
- Création d'une description d'unité
- Création des informations complémentaires CPI-C
- Ajout d'un poste communications pour APPC
- Ajouter une entrée de liste de configuration

Procédure

1. Configurez le noeud local en créant une description de ligne et en ajoutant un poste de routage.

a) Créez une description de ligne.

Si la description de ligne n'a pas encore été créée, utilisez la commande **CRTLINTRN** pour spécifier des valeurs pour **Description de ligne** (6) et **Nom de ressource** (7), comme illustré dans l'exemple suivant:

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

Line description . . . . . TOKENRINGL Name
Resource name . . . . . LIN041 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) Ajoutez un poste de routage.

Entrez la commande **ADDRTGE** et appuyez sur Entrée, puis indiquez votre propre valeur pour **Subsystem description** (5), ainsi que les valeurs indiquées dans l'exemple suivant pour **Routing entry sequence number**, **Compare value** (8), **Starting position**, **Program to call** et **Library** contenant le programme à appeler.

```

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.
+

```

Démarrez le sous-système en entrant la commande STRSBS *subsystem description* (5) et en appuyant sur Entrée.

2. Créez la connexion au noeud partenaire en créant une description de contrôleur, une description d'unité et les informations complémentaires CPI-C, et en ajoutant un poste communications pour APPC, ainsi qu'un poste liste de configuration.



Cet exemple concerne une connexion à un système Windows , mais les étapes sont les mêmes pour les autres noeuds.

a) Création d'une description de contrôleur

Sur une ligne de commande, tapez CRTCTLAPPC et appuyez sur Entrée, puis indiquez des valeurs pour **Controller description** (12), définissez **Link type** sur *LANet définissez **Online at IPL** sur *NO.

```

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.
+

```

Appuyez deux fois sur Entrée, puis sur F10, puis spécifiez les valeurs pour **Switched line list** (6), **Remote network identifier** (9), **Remote control point** (10) et **LAN remote adapter address** (16) et appuyez sur Entrée.

```

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) Créez une description d'unité.

Tapez la commande CRTDEVAPPC et appuyez sur Entrée, puis spécifiez les valeurs pour **Description de l'unité (13)**, **Emplacement distant (11)**, **Emplacement local (3)**, **Identificateur de réseau distant (9)** et **Contrôleur connecté (12)**.

```

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.      +

```

Remarque : Vous pouvez éviter d'avoir à créer manuellement des descriptions de contrôleur et d'unité en tirant parti du service de configuration automatique d'IBM i. Pour plus d'informations, reportez-vous à la documentation IBM i.

c) Créez les informations complémentaires CPI-C.

Tapez CRTCSI et appuyez sur F10, puis indiquez des valeurs pour **Side information (14)**, **Remote location (11)**, **Transaction program (15)**, **Local location (3)**, **Modeet Remote network identifier (9)** et appuyez sur Entrée.

```

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) Ajoutez un poste communications pour APPC.

Sur une ligne de commande, tapez ADDCMNE et appuyez sur Entrée, puis indiquez des valeurs pour **Subsystem description** (5) et **Device** (13), puis appuyez à nouveau sur Entrée.

```

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 SBSDB required.

```

e) Ajoutez une entrée de liste de configuration.

Tapez ADDCFGLE *APPNRMT et appuyez sur F4, puis indiquez des valeurs pour **Remote location name** (11), **Remote network identifier** (9), **Local location name** (3), **Remote control point** (10) et **Control point net ID** (9) et appuyez sur Entrée.

```

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

```

Que faire ensuite

La connexion LU 6.2 est maintenant établie. Vous êtes prêt à effectuer la configuration comme décrit dans «[Configuration des canaux sur IBM i](#)», à la page 28.

Tâches associées

«[Etablissement d'une connexion TCP sous IBM i](#)», à la page 26

Si TCP est déjà configuré, il n'y a pas de tâches de configuration supplémentaires. Si TCP/IP n'est pas configuré, vous devez ajouter une interface TCP/IP, ajouter une interface de bouclage TCP/IP et ajouter une route par défaut.

IBM i Paramètres de configuration d'une connexion LU 6.2 sous IBM i

Paramètres nécessaires pour configurer la communication entre IBM MQ sur un système IBM i et l'une des autres plateformes IBM MQ à l'aide d'une connexion LU 6.2 .

Utilisez ces tables avec les tables de la plateforme à laquelle vous vous connectez.

Lorsque des nombres apparaissent dans la colonne *Référence* , ils indiquent que la valeur doit correspondre à celle du tableau approprié ailleurs dans cette section. Les étapes de tâche du «[Etablissement d'une connexion LU 6.2 sous IBM i](#)», à la page 16 font référence aux valeurs de la colonne *ID* de ce tableau.

Les entrées de la colonne *Nom du paramètre* sont expliquées dans «[Explication des termes utilisés dans les tableaux](#)», à la page 24.

Définition du noeud local

Tableau 7. Exemples de configuration pour la définition du noeud local			
division d'identification	Nom du paramètre :	Référence	Exemple utilisé
1	ID réseau local		identificateur de réseau
2	Nom du point de contrôle local		AS400PU
3	Nom de LU		AS400LU

Tableau 7. Exemples de configuration pour la définition du noeud local (suite)

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
4	Adresse de destination du réseau local		10005A5962EF
5	Description du sous-système		QCMN
6	Description de ligne		TOKENRINGL
7	Nom de la ressource		LIN041
8	Nom du programme de transaction locale		MQSERIES

Connexion à IBM MQ on Windows

Windows

Windows

Tableau 8. Exemples de configuration pour la connexion à IBM MQ sous Windows

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
9	ID réseau	2	identificateur de réseau
10	Nom du point de contrôle	3	WINNTCP
11	Nom de LU	5	WINNTLU
12	Description du contrôleur		WINNTCP
13	Unité		WINNTLU
14	Informations complémentaires		NTCPIC
15	Programme transactionnel	7	MQSERIES
16	Adresse de la carte de réseau local	9	08005AA5FAB9
17	Mode	17	#INTER

Connexion à IBM MQ sur AIX

AIX

Tableau 9. Exemples de configuration pour la connexion à IBM MQ sur un système AIX

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
9	ID réseau	1	identificateur de réseau
10	Nom du point de contrôle	2	AIXUnité physique

Tableau 9. Exemples de configuration pour la connexion à IBM MQ sur un système AIX (suite)

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
11	Nom de LU	4	AIXLU
12	Description du contrôleur		AIXUnité physique
13	Unité		AIXLU
14	Informations complémentaires		CPICAIX
15	Programme transactionnel	6	MQSERIES
16	Adresse de la carte de réseau local	8	123456789012
17	Mode	14	#INTER

Connexion à IBM MQ sur Linux (plateformex86)

Linux

Tableau 10. Exemples de configuration pour la connexion à IBM MQ sous Linux (plateformex86)

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
9	ID réseau	4	identificateur de réseau
10	Nom du point de contrôle	2	LINUXPU
11	Nom de LU	5	LINUXLU
12	Description du contrôleur		LINUXPU
13	Unité		LINUXLU
14	Informations complémentaires		LXCPIIC
15	Programme transactionnel	7	MQSERIES
16	Adresse de la carte de réseau local	8	08005AC6DF33
17	Mode	6	#INTER

connexion à IBM MQ for z/OS

z/OS

Tableau 11. Exemples de configuration pour la connexion à IBM MQ for z/OS

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
9	ID réseau	2	identificateur de réseau

Tableau 11. Exemples de configuration pour la connexion à IBM MQ for z/OS (suite)

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
10	Nom du point de contrôle	3	MVSPU
11	Nom de LU	4	MVSLU
12	Description du contrôleur		MVSPU
13	Unité		MVSLU
14	Informations complémentaires		MVSCPIC
15	Programme transactionnel	7	MQSERIES
16	Adresse de la carte de réseau local	8	400074511092
17	Mode	6	#INTER

Connexion à un système VSE/ESA

Tableau 12. Exemples de configuration pour la connexion à un système VSE/ESA

division d'identification	Parameter Name	Référence	Exemple utilisé
9	ID réseau	1	identificateur de réseau
10	Nom du point de contrôle	2	VSEPU
11	Nom de LU	3	VSELU
12	Description du contrôleur		VSEPU
13	Unité		VSELU
14	Informations complémentaires		VSECPIC
15	Programme transactionnel	4	MQ01
16	Adresse de la carte de réseau local	5	400074511092
17	Mode		#INTER

Explication des termes utilisés dans les tableaux

1 2 3

Pour plus d'informations sur la recherche des valeurs configurées, voir [«Comment rechercher des attributs de réseau»](#), à la page 25.

Adresse de destination du réseau local 4

Adresse matérielle de l'adaptateur de réseau en anneau à jeton du système IBM i . Vous pouvez trouver la valeur à l'aide de la commande `DSPLIND Line description (6)`.

5 Description de sous-système

Ce paramètre correspond au nom de tout sous-système IBM i actif lors de l'utilisation du gestionnaire de files d'attente. Le nom QCMN a été utilisé car il s'agit du sous-système de communication IBM i .

6 Description de ligne

Si ce paramètre a été indiqué, il est indiqué dans la zone Description du nom de la ressource. Pour plus d'informations, voir «Comment trouver la valeur du nom de ressource», à la page 26. Si la valeur n'est pas indiquée, vous devez créer une description de ligne.

7 Nom de la ressource

Pour plus d'informations sur la recherche de la valeur configurée, voir «Comment trouver la valeur du nom de ressource», à la page 26.

8 Nom du programme de transaction locale

Les applications IBM MQ qui tentent de converser avec ce poste de travail spécifient un nom symbolique pour le programme à exécuter à l'extrémité réceptrice. Ce nom est défini sur la définition de canal au niveau de l'émetteur. Par souci de simplicité, dans la mesure du possible, utilisez le nom de programme de transaction MQSERIES, ou dans le cas d'une connexion à VSE/ESA, dont la longueur est limitée à 4 octets, utilisez MQTP.

Pour plus d'informations, voir Paramètres sur le système IBM i local pour une plateforme de gestionnaire de files d'attente éloignées .

12 Description du contrôleur

Ce paramètre est un alias pour le nom de point de contrôle (ou nom de noeud) du système partenaire. Pour des raisons de commodité, nous avons utilisé le nom réel du partenaire dans cet exemple.

13 périphériques

Ce paramètre est un alias pour l'unité logique du système partenaire. Pour des raisons de commodité, nous avons utilisé le nom de LU du partenaire dans cet exemple.

14 Informations complémentaires

Ce paramètre est le nom donné au profil d'informations côté CPI-C. Vous indiquez votre propre nom de 8 caractères.

Comment rechercher des attributs de réseau

Le noeud local a été partiellement configuré dans le cadre de l'installation de IBM i . Pour afficher les attributs de réseau en cours, entrez la commande **DSPNETA**.

Si vous devez modifier ces valeurs, utilisez la commande **CHGNETA**. Un IPL peut être nécessaire pour appliquer vos modifications.

```
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
```

Vérifiez que les valeurs de **Local network ID** (1), **Local control point name** (2) et **Default local location** (3) correspondent aux valeurs de la table ou à vos propres valeurs si vous les avez modifiées.

Comment trouver la valeur du nom de ressource

Pour trouver la valeur du nom de ressource, entrez WRKHDWRSC TYPE(*CMN) et appuyez sur Entrée.

Le panneau Gestion des ressources de communication s'affiche. La valeur de **Nom de ressource** est trouvée en tant que port de l'anneau à jeton. Il s'agit de LIN041 dans cet exemple.

```
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
```

Etablissement d'une connexion TCP sous IBM i

Si TCP est déjà configuré, il n'y a pas de tâches de configuration supplémentaires. Si TCP/IP n'est pas configuré, vous devez ajouter une interface TCP/IP, ajouter une interface de bouclage TCP/IP et ajouter une route par défaut.

Procédure

1. Ajoutez une interface TCP/IP.

Sur une ligne de commande, entrez ADDTCPIFC et appuyez sur Entrée, puis indiquez l' **adresse IP** et la **description de ligne**, ainsi qu'un **masque de sous-réseau** de la machine, puis appuyez à nouveau sur Entrée.

```

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. Ajoutez une interface de bouclage TCP/IP.

Sur la ligne de commande, tapez ADDTCPIFC et appuyez sur Entrée, puis indiquez les valeurs pour **Adresse IP, Description de ligne et Masque de sous-réseau.**

```

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. Ajoutez une route par défaut.

Sur la ligne de commande, tapez ADDTCP RTE et appuyez sur Entrée, puis indiquez les valeurs appropriées à votre réseau et appuyez sur Entrée pour créer une entrée de route par défaut.

```

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.

```

Que faire ensuite

La connexion TCP est maintenant établie. Vous êtes prêt à effectuer la configuration comme décrit dans «[Configuration des canaux sur IBM i](#)», à la page 28.

Tâches associées

«[Etablissement d'une connexion LU 6.2 sous IBM i](#)», à la page 16

Pour établir une connexion LU 6.2 sous IBM i, vous devez configurer le nœud local et le connecter à un nœud partenaire.

Configuration des canaux sur IBM i

Pour configurer IBM MQ pour l'exemple de configuration sous IBM i, effectuez les étapes de configuration de base pour le gestionnaire de files d'attente, puis configurez les canaux émetteur et récepteur.

Pourquoi et quand exécuter cette tâche

La commande **WRKMQMQ** permet d'afficher le menu de configuration IBM MQ .

Démarrez le programme d'écoute de canal TCP à l'aide de la commande **STRMQMLSR**.

Démarrez un canal émetteur à l'aide de la commande **STRMQMCHL** *CHLNAME(channel_name)*.

Remarque : Les erreurs AMQ* sont placées dans le journal relatif au travail qui a détecté l'erreur. La commande **WRKACTJOB** permet d'afficher la liste des travaux. Sous le nom de sous-système QSYSWRK, localisez le travail et entrez 5 pour qu'il soit utilisé avec ce travail. Les journaux IBM MQ sont préfixés AMQ.

Procédure

1. Créez un gestionnaire de files d'attente.
 - a) Tapez CRTMQM et appuyez sur Entrée.

```

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) Dans la zone **Nom du gestionnaire de files d'attente de messages** , entrez AS400 et dans la zone **File d'attente de messages non distribuée** , entrez DEAD . LETTER . QUEUE , puis appuyez sur Entrée.
- c) Démarrez le gestionnaire de files d'attente en entrant STRMQM MQMNAME (AS400).
- d) Créez la file d'attente de messages non distribuée à l'aide des paramètres suivants:

```

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

```

Pour plus d'informations et un exemple de définition d'une file d'attente, voir l'étape [«2»](#), à la page [29](#).

2. Définissez une file d'attente.

- a) Entrez CRTMQMQ sur la ligne de commande.

```

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) Renseignez les deux zones de ce panneau et appuyez sur Entrée.

Un autre panneau s'affiche, avec des zones d'entrée pour les autres paramètres dont vous disposez. Les valeurs par défaut peuvent être utilisées pour tous les autres attributs de file d'attente.

3. Définissez un canal.

a) Entrez CRTMQMCHL sur la ligne de commande pour afficher le panneau **Create MQM Channel**.

```

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) Renseignez les deux zones de ce panneau et appuyez sur Entrée.

Un autre panneau s'affiche alors pour vous permettre de spécifier les valeurs des autres paramètres indiqués précédemment. Les valeurs par défaut peuvent être utilisées pour tous les autres attributs de canal.

4. Configurez les canaux pour l'exemple de configuration.

Pour plus d'informations sur les paramètres utilisés dans les exemples suivants, voir [«Paramètres de configuration de canal pour IBM i»](#), à la page 31.

Windows Ces exemples permettent de connecter IBM MQ sous IBM i à IBM MQ sous Windows. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs appropriées des tables de [«Paramètres de configuration de canal pour IBM i»](#), à la page 31 à la place des valeurs de Windows.

a) Définissez le canal émetteur comme illustré dans les exemples suivants:

- Utilisation de SNA:

```

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

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

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

```

- Utilisation de TCP:

```

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) Définissez le canal récepteur comme illustré dans les exemples suivants:

- Utilisation de SNA:

```

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

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

```

- Utilisation de TCP:

```

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

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

```

Paramètres de configuration de canal pour IBM i

Paramètres requis pour configurer les canaux pour l'exemple de configuration sur IBM i.

L'étape «4», à la page 30 de «Configuration des canaux sur IBM i», à la page 28 décrit la configuration à effectuer sur le gestionnaire de files d'attente IBM i pour implémenter le canal décrit dans «Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5. Les exemples dans «Configuration des canaux sur IBM i», à la page 28 permettent de connecter IBM MQ for IBM i et IBM MQ for Windows. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs de la table appropriée à la place des valeurs de Windows.

Remarque :

1. Les mots en **gras** sont des valeurs suggérées et reflètent les noms des objets IBM MQ utilisés dans ces exemples. Vous pouvez les modifier dans l'installation de votre produit mais, si vous le faites, veillez à utiliser vos propres valeurs lors de l'utilisation des exemples de cette section.
2. La commande IBM MQ channel ping (**PNGMQMCHL**) s'exécute de manière interactive, alors que le démarrage d'un canal entraîne la soumission d'un travail par lots. Si une commande ping de canal aboutit mais que le canal ne démarre pas, les définitions de réseau et IBM MQ sont probablement correctes, mais que l'environnement IBM i pour le travail par lots ne l'est pas. Par exemple, assurez-vous que QSYS2 est inclus dans la partie système de la liste des bibliothèques et pas seulement dans votre liste de bibliothèques personnelle.

Pour plus d'informations et des exemples de création des objets répertoriés dans les tableaux, voir «Configuration des canaux sur IBM i», à la page 28.

Définition du noeud local

Tableau 13. Exemples de configuration pour la définition du noeud local

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
A	Nom du gest. de files		AS400
B	Nom de la file d'attente locale		AS400.LOCALQ

Connexion à IBM MQ sur Windows

Windows

Les valeurs de cette section du tableau doivent correspondre aux valeurs utilisées dans «Paramètres de configuration de canal pour Windows», à la page 46, comme indiqué.

Tableau 14. Exemples de configuration pour la connexion à IBM MQ sous Windows

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	winnt
D	Nom de la file d'attente distante		WINNT.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	WINNT.LOCALQ
F	Nom de la file d'attente de transmission		winnt
G	Nom du canal émetteur (SNA)		AS400.WINNT.SNA
H	Nom du canal émetteur (TCP/IP)		AS400.WINNT.TCP
I	Nom de canal récepteur (SNA)	G	WINNT.AS400.SNA
J	Nom du canal récepteur (TCP/IP)	H	WINNT.AS400.TCP

Connexion à IBM MQ sur AIX

AIX

Les valeurs de cette section du tableau doivent correspondre aux valeurs utilisées dans «Paramètres de configuration de canal pour AIX», à la page 13, comme indiqué.

Tableau 15. Exemples de configuration pour la connexion à IBM MQ sous AIX

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	AIX
D	Nom de la file d'attente distante		AIX:REMOTEQ
E	Nom de la file d'attente sur le système distant	B	AIX.LOCALQ

Tableau 15. Exemples de configuration pour la connexion à IBM MQ sous AIX (suite)

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
F	Nom de la file d'attente de transmission		AIX
G	Nom du canal émetteur (SNA)		AS400.AIX.SNA
H	Nom du canal émetteur (TCP/IP)		AS400.AIX.TCP
I	Nom de canal récepteur (SNA)	G	AIX.AS400.SNA
J	Nom du canal récepteur (TCP)	H	AIX.AS400.TCP

Connexion à IBM MQ sur Linux

Linux

Les valeurs de cette section du tableau doivent correspondre aux valeurs utilisées dans «Paramètres de configuration de canal pour Linux», à la page 39, comme indiqué.

Tableau 16. Exemples de configuration pour la connexion à IBM MQ sous Linux

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	Linux
D	Nom de la file d'attente distante		LINUX.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	LINUX.LOCALQ
F	Nom de la file d'attente de transmission		Linux
G	Nom du canal émetteur (SNA)		AS400.LINUX.SNA
H	Nom du canal émetteur (TCP/IP)		AS400.LINUX.TCP
I	Nom de canal récepteur (SNA)	G	LINUX.AS400.SNA
J	Nom du canal récepteur (TCP/IP)	H	LINUX.AS400.TCP

connexion à IBM MQ for z/OS

z/OS

Les valeurs de cette section du tableau doivent correspondre aux valeurs utilisées dans «Channel configuration parameters for z/OS», à la page 50, comme indiqué.

Tableau 17. Exemples de configuration pour la connexion à IBM MQ for z/OS

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	système MVS

Tableau 17. Exemples de configuration pour la connexion à IBM MQ for z/OS (suite)

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
D	Nom de la file d'attente distante		MVS.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	MVS.LOCALQ
F	Nom de la file d'attente de transmission		système MVS
G	Nom du canal émetteur (SNA)		AS400.MVS.SNA
H	Nom du canal émetteur (TCP)		AS400.MVS.TCP
I	Nom de canal récepteur (SNA)	G	MVS.AS400.SNA
J	Nom du canal récepteur (TCP)	H	MVS.AS400.TCP

Connexion à un système VSE/ESA

Les valeurs de cette section du tableau doivent correspondre aux valeurs utilisées dans votre système VSE/ESA .

Tableau 18. Exemples de configuration pour la connexion à un système VSE/ESA

divisi on d'iden tificati on	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	système d'exploitation à mémoire virtuelle améliorée
D	Nom de la file d'attente distante		VSE.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	VSE.LOCALQ
F	Nom de la file d'attente de transmission		système d'exploitation à mémoire virtuelle améliorée
G	Nom du canal émetteur		AS400.VSE.SNA
I	Nom du canal récepteur	G	VSE.AS400.SNA

Linux

Exemple: configuration de la communication multiplateforme IBM MQ sous Linux

Cet exemple montre comment configurer des liaisons de communication entre IBM MQ sur Linux et IBM MQ sur une autre plateforme et établir un canal de travail vers cette plateforme.

Avant de commencer

Pour plus d'informations sur cet exemple et sur la façon de l'utiliser, voir [«Exemple: configuration de la communication multiplateforme pour IBM MQ»](#), à la page 5 et [«Comment utiliser les exemples de communication multiplateforme»](#), à la page 7.

Pourquoi et quand exécuter cette tâche

Cet exemple décrit la configuration de la communication multiplateforme entre IBM MQ sur Linux et les plateformes suivantes:

-  Windows
-  AIX
-  IBM i
-  z/OS

`MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.

Etablissez une connexion réseau à l'aide de LU 6.2 ou TCP.

Remarque : Pour TCP, certaines distributions Linux utilisent désormais le démon inet étendu (XINETD) à la place du démon inet (INETD). Les instructions suivantes vous indiquent comment établir une connexion TCP à l'aide du démon inet ou du démon inet étendu.

Procédure

1. Etablissement d'une connexion réseau à l'aide de LU6.2

Remarque : Les informations de cette section s'appliquent uniquement à IBM MQ for Linux (plateformex86). Elle ne s'applique pas à IBM MQ for Linux (plateformex86-64), IBM MQ for Linux (plateforme zSeries s390x) ou IBM MQ for Linux (plateforme Power).

Pour obtenir les informations les plus récentes sur la configuration de SNA sur TCP/IP, reportez-vous au Guide d'administration de votre version de Linux dans la documentation suivante: [Communications Server for Data Center Deployment on Linux library](#).

2. Etablir une connexion TCP à l'aide du démon inet (INETD)

- a) Editez le fichier `/etc/services`.

Si le fichier ne contient pas la ligne suivante, ajoutez-la comme suit:

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

Remarque : Pour éditer ce fichier, vous devez être connecté en tant que superutilisateur ou superutilisateur.

- b) Editez le fichier `/etc/inetd.conf`.

Si vous ne disposez pas de la ligne suivante dans ce fichier, ajoutez-la comme suit:

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

- c) Recherchez l'ID de processus de l'ID inetd à l'aide de la commande suivante:

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

- d) Exécutez la commande :

```
kill -1 inetd processid
```

Si vous disposez de plusieurs gestionnaires de files d'attente sur votre système et que vous avez donc besoin de plusieurs services, vous devez ajouter une ligne pour chaque gestionnaire de files d'attente supplémentaire à `/etc/services` et à `inetd.conf`.

Exemple :

```
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
```

Cela permet d'éviter la génération de messages d'erreur en cas de limitation du nombre de demandes de connexion en attente mises en file d'attente sur un port TCP unique. Pour plus d'informations sur le nombre de demandes de connexion en attente, voir [Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP](#).

Le processus inetd sous Linux peut limiter le débit des connexions entrantes sur un port TCP. La valeur par défaut est de 40 connexions dans un intervalle de 60 secondes. Si vous avez besoin d'un débit plus élevé, spécifiez une nouvelle limite sur le nombre de connexions entrantes dans un intervalle de 60 secondes en ajoutant un point (.) suivi de la nouvelle limite au paramètre nowait du service approprié dans inetd.conf. Par exemple, pour une limite de 500 connexions dans un intervalle de 60 secondes, utilisez:

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

`MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.

3. Etablir une connexion TCP à l'aide du démon inet étendu (XINETD)

Les instructions suivantes décrivent comment le démon inet étendu est implémenté sur Red Hat Linux. Si vous utilisez une autre distribution Linux, vous devrez peut-être adapter ces instructions.

a) Editez le fichier `/etc/services`.

Si le fichier ne contient pas la ligne suivante, ajoutez-la comme suit:

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

Si le fichier ne contient pas la ligne suivante, ajoutez-la comme suit:

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

b) Créez un fichier appelé IBM MQ dans le répertoire de configuration XINETD, `/etc/xinetd.d`, en ajoutant la strophe suivante au fichier:

```
# 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) Redémarrez le démon inet étendu en exécutant la commande suivante:

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

Si votre système comporte plusieurs gestionnaires de files d'attente et que vous avez besoin de plusieurs services, vous devez ajouter une ligne à `/etc/services` pour chaque gestionnaire de files d'attente supplémentaire. Vous pouvez créer un fichier dans le répertoire `/etc/xinetd.d` pour chaque service ou ajouter des sections supplémentaires au fichier IBM MQ que vous avez créé précédemment.

Le processus xinetd sur Linux peut limiter le débit des connexions entrantes sur un port TCP. La valeur par défaut est de 50 connexions dans un intervalle de 10 secondes. Si vous avez besoin d'un débit plus élevé, spécifiez une nouvelle limite sur le débit des connexions entrantes en spécifiant l'attribut 'cps' dans le fichier de configuration xinetd. Par exemple, pour une limite de 500 connexions dans un intervalle de 60 secondes, utilisez:

```
cps = 500 60
```

4. Terminez la configuration maintenant que la connexion TCP/IP est établie.
Accédez à [«Configuration des canaux sur Linux»](#), à la page 37.

Linux Configuration des canaux sur Linux

Pour configurer IBM MQ pour l'exemple de configuration sous Linux, effectuez les étapes de configuration de base pour le gestionnaire de files d'attente, puis configurez les canaux émetteur et récepteur.

Avant de commencer

Avant de commencer le processus, vérifiez que vous avez d'abord créé l'ID utilisateur mqm et le groupe mqm, puis définissez le mot de passe.

Démarrez n'importe quel canal à l'aide de la commande suivante:

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

Pourquoi et quand exécuter cette tâche

Remarques :

1. Les exemples de programme sont installés dans `MQ_INSTALLATION_PATH/samp`, où `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.
2. Les journaux d'erreurs sont stockés dans les erreurs `/var/mqm/qmgrs/ qmgrname /`.
3. Lorsque vous utilisez l'interpréteur de commandes **runmqsc** pour entrer des commandes d'administration, un signe + à la fin d'une ligne indique que la ligne suivante est une continuation. Assurez-vous qu'il existe un espace entre le dernier paramètre et le caractère de continuation.

Procédure

1. Configurez la configuration de base:
 - a) Créez le gestionnaire de files d'attente et un ensemble d'objets par défaut à partir de l'invite UNIX , à l'aide de la commande suivante:

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

où :

linux

Est le nom du gestionnaire de files d'attente

-q

Indique qu'il doit devenir le gestionnaire de files d'attente par défaut

-u dlqname

Indique le nom de la file d'attente de rebut

- b) Démarrez le gestionnaire de files d'attente à partir de l'invite UNIX à l'aide de la commande suivante:

```
strmqm linux
```

où *linux* est le nom donné au gestionnaire de files d'attente lors de sa création.

2. Configurez les canaux pour l'exemple de configuration.

Pour plus d'informations sur les paramètres utilisés dans les exemples suivants, voir «Paramètres de configuration de canal pour Linux», à la page 39. Dans chaque cas, l'exemple montre la commande MQSC. Démarrez **runmqsc** à partir d'une ligne de commande Linux et entrez chaque commande à tour de rôle, ou générez les commandes dans un fichier de commandes.

Windows Ces exemples permettent de connecter IBM MQ sous Linux à IBM MQ sous Windows. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs appropriées des tables de «Paramètres de configuration de canal pour Linux», à la page 39 à la place des valeurs de Windows.

a) Définissez le canal émetteur comme illustré dans les exemples suivants:

- Utilisation de SNA

```
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
  trptype(lu62) +
  conname('WINNTCPIC') +                       14
  xmitq(WINNT) +                               F
  replace
```

- A l'aide de TCP

```
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
  trptype(tcp) +
  conname(remote_tcpip_hostname) +
  xmitq(WINNT) +                               F
  replace
```

b) Définition du canal récepteur, comme illustré dans les exemples suivants:

- Utilisation de SNA:

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

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

- Utilisation de TCP:

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

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

Paramètres de configuration de canal pour Linux

Paramètres requis pour configurer les canaux pour l'exemple de configuration sur Linux.

L'étape «2», à la page 38 de «Configuration des canaux sur Linux», à la page 37 décrit la configuration à effectuer sur le gestionnaire de files d'attente Linux pour implémenter le canal décrit dans «Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5. Les exemples dans «Configuration des canaux sur Linux», à la page 37 permettent de connecter IBM MQ for IBM i et IBM MQ for Windows. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs de la table appropriée à la place des valeurs de Windows.

Remarque : Les mots en **gras** sont des valeurs suggérées et reflètent les noms des objets IBM MQ utilisés dans ces exemples. Vous pouvez les modifier dans l'installation de votre produit mais, si vous le faites, veillez à utiliser vos propres valeurs lorsque vous utilisez les exemples de cette section.

Définition du noeud local

Tableau 19. Exemples de configuration pour la définition du noeud local

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
A	Nom du gest. de files		Linux
B	Nom de la file d'attente locale		LINUX.LOCALQ

Connexion à IBM MQ sur Windows

Windows

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour Windows», à la page 46, comme indiqué.

Tableau 20. Exemples de configuration pour la connexion à IBM MQ sous Windows

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	winnt
D	Nom de la file d'attente distante		WINNT.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	WINNT.LOCALQ
F	Nom de la file d'attente de transmission		winnt
G	Nom du canal émetteur (SNA)		LINUX.WINNT.SNA
H	Nom du canal émetteur (TCP/IP)		LINUX.WINNT.TCP
I	Nom de canal récepteur (SNA)	G	WINNT.LINUX.SNA
J	Nom du canal récepteur (TCP)	H	WINNT.LINUX.TCP

Connexion à IBM MQ sur AIX

AIX

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Configuration des canaux sur AIX», à la page 10, comme indiqué.

Tableau 21. Exemples de configuration pour la connexion à IBM MQ sous AIX

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	AIX
D	Nom de la file d'attente distante		AIX.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	AIX.LOCALQ
F	Nom de la file d'attente de transmission		AIX
G	Nom du canal émetteur (SNA)		.LINUX.AIX.SNA
H	Nom du canal émetteur (TCP/IP)		LINUX.AIX.TCP
I	Nom de canal récepteur (SNA)	G	AIX.LINUX.SNA
J	Nom du canal récepteur (TCP/IP)	H	AIX.LINUX.TCP

connexion à IBM MQ for IBM i

IBM i

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour IBM i», à la page 31, comme indiqué.

Tableau 22. Exemples de configuration pour la connexion à IBM MQ sous IBM i

division d'identification	Parameter Name	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	AS400
D	Nom de la file d'attente distante		AS400.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	AS400.LOCALQ
F	Nom de la file d'attente de transmission		AS400
G	Nom du canal émetteur (SNA)		LINUX.AS400.SNA
H	Nom du canal émetteur (TCP)		LINUX.AS400.TCP
I	Nom de canal récepteur (SNA)	G	AS400.LINUX.SNA
J	Nom du canal récepteur (TCP)	H	AS400.LINUX.TCP

connexion à IBM MQ for z/OS

z/OS

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Channel configuration parameters for z/OS», à la page 50, comme indiqué.

Tableau 23. Exemples de configuration pour la connexion à IBM MQ for z/OS

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	 système MVS
D	Nom de la file d'attente distante		MVS.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	MVS.LOCALQ
F	Nom de la file d'attente de transmission		 système MVS
G	Nom du canal émetteur (SNA)		LINUX.MVS.SNA
H	Nom du canal émetteur (TCP)		LINUX.MVS.TCP
I	Nom de canal récepteur (SNA)	G	MVS.LINUX.SNA
J	Nom du canal récepteur (TCP)	H	MVS.LINUX.TCP

Connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente



Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Shared channel configuration parameters», à la page 59, comme indiqué.

Tableau 24. Exemples de configuration pour la connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	QSG
D	Nom de la file d'attente distante		QSG.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	QSG.SHAREDQ
F	Nom de la file d'attente de transmission		QSG
G	Nom du canal émetteur (SNA)		LINUX.QSG.SNA
H	Nom du canal émetteur (TCP)		LINUX.QSG.TCP
I	Nom de canal récepteur (SNA)	G	QSG.LINUX.SNA
J	Nom du canal récepteur (TCP)	H	QSG.LINUX.TCP

Windows Exemple: configuration de la communication multiplateforme IBM MQ sous Windows

Cet exemple montre comment configurer des liaisons de communication entre IBM MQ sur Windows et IBM MQ sur une autre plateforme et établir un canal de travail vers cette plateforme.

Avant de commencer

Pour plus d'informations sur cet exemple et sur la façon de l'utiliser, voir «Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5 et «Comment utiliser les exemples de communication multiplateforme», à la page 7.

Pourquoi et quand exécuter cette tâche

Cet exemple décrit la configuration de la communication multiplateforme entre IBM MQ sur Windows et les plateformes suivantes:

-  AIX
-  IBM i
-  Linux
-  z/OS

Procédure

1. Etablissez une connexion réseau à l'aide de LU6.2.

Pour plus d'informations sur la configuration de AnyNet SNA sur TCP/IP, voir [AnyNet SNA sur TCP/IP et Communications Server for Windows](#) .

2. Etablissez une connexion réseau à l'aide de TCP.

La pile TCP fournie avec les systèmes Windows n'inclut pas de démon *inet* ou équivalent.

Vous devez démarrer le programme d'écoute de manière explicite avant de démarrer les canaux. Il permet aux canaux de réception de démarrer automatiquement en réponse à une demande provenant d'un canal d'envoi entrant.

Utilisez la commande suivante pour démarrer le programme d'écoute TCP IBM MQ :

```
runmqtsr -t tcp
```

3. Etablissez une connexion réseau à l'aide de NetBIOS.

- a) A chaque extrémité du canal, indiquez le nom NetBIOS local à utiliser par les processus de canal IBM MQ dans le fichier de configuration du gestionnaire de files d'attente qm.ini.

Par exemple, la section NETBIOS dans Windows à la fin de l'envoi peut se présenter comme suit:

```
NETBIOS:  
LocalName=WNTNETB1
```

et à la fin de la réception, se présenter comme suit:

```
NETBIOS:  
LocalName=WNTNETB2
```

Chaque processus IBM MQ doit utiliser un nom NetBIOS local différent. N'utilisez pas votre nom de système comme nom NetBIOS car Windows l'utilise déjà.

- b) A chaque extrémité du canal, vérifiez le numéro de carte de réseau local utilisé sur votre système.

La valeur par défaut de IBM MQ for Windows pour l'adaptateur logique numéro 0 est NetBIOS exécuté sur un réseau Internet Protocol . Pour utiliser l'adaptateur NetBIOS natif, vous devez sélectionner l'adaptateur logique numéro 1. Voir [Etablissement du numéro de carte de réseau local](#).

Indiquez le numéro de carte de réseau local correct dans la section NETBIOS du registre Windows .
Exemple :

```
NETBIOS:  
AdapterNum=1
```

- c) Pour que l'initialisation du canal émetteur fonctionne, spécifiez le nom NetBIOS local à l'aide de la variable d'environnement MQNAME:

```
SET MQNAME=WNTNETB1I
```

Remarque : Il doit être unique.

- d) A l'extrémité émettrice, définissez un canal spécifiant le nom NetBIOS utilisé à l'autre extrémité du canal.

Exemple :

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

Vous devez spécifier l'option MCATYPE (THREAD) car, sous Windows, les canaux émetteurs doivent être exécutés en tant qu'unités d'exécution.

- e) A la réception, définissez le canal récepteur correspondant.

Exemple :

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

- f) Lancez l'initiateur de canal.

Chaque nouveau canal est démarré en tant qu'unité d'exécution plutôt qu'en tant que nouveau processus:

```
runmqchi
```

- g) A la fin de la réception, démarrez le programme d'écoute IBM MQ :

```
runmqclsr -t netbios
```

Vous pouvez éventuellement spécifier des valeurs pour le nom du gestionnaire de files d'attente, le nom local NetBIOS, le nombre de sessions, le nombre de noms et le nombre de commandes. Pour plus d'informations sur la configuration des connexions NetBIOS, voir [Définition d'une connexion NetBIOS sur Windows](#).

4. Terminez la configuration maintenant que la connexion réseau est établie. Voir [«Configuration des canaux sur Windows»](#), à la page 43.

Windows Configuration des canaux sur Windows

Pour configurer IBM MQ pour l'exemple de configuration sous Windows, effectuez les étapes de configuration de base pour le gestionnaire de files d'attente, puis configurez les canaux émetteur et récepteur.

Pourquoi et quand exécuter cette tâche

Remarques :

1. Vous pouvez utiliser l'exemple de programme AMQSBCG pour afficher le contenu et les en-têtes de tous les messages d'une file d'attente. Exemple :

```
AMQSBCG q_name qmgr_name
```

affiche le contenu de la file d'attente *q_name* définie dans le gestionnaire de files d'attente *qmgr_name*.

Vous pouvez également utiliser le navigateur de messages dans l'explorateur IBM MQ .

2. Vous pouvez démarrer n'importe quel canal à partir de l'invite de commande à l'aide de la commande

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

3. Les journaux d'erreurs se trouvent dans les répertoires *MQ_INSTALLATION_PATH\qmgrs\qmgrname\errors* et *MQ_INSTALLATION_PATH\qmgrs\@system\errors*. Dans les deux cas, les messages les plus récents se trouvent à la fin de *amqerr01.log*.

MQ_INSTALLATION_PATH représente le répertoire de haut niveau dans lequel IBM MQ est installé.

4. Lorsque vous utilisez l'interpréteur de commandes **runmqsc** pour entrer des commandes d'administration, un signe + à la fin d'une ligne indique que la ligne suivante est une continuation. Assurez-vous qu'il existe un espace entre le dernier paramètre et le caractère de continuation.

Procédure

1. Pour définir la configuration de base à l'aide de l'invite de commande, procédez comme suit:

- a) Créez le gestionnaire de files d'attente et un ensemble d'objets par défaut à l'aide de la commande suivante:

```
crtmqm -u dlqname -q winnt
```

où :

winnt

Est le nom du gestionnaire de files d'attente

-q

Indique qu'il doit devenir le gestionnaire de files d'attente par défaut

-u dlqname

Indique le nom de la file d'attente de messages non distribuables

- b) Démarrez le gestionnaire de files d'attente à l'aide de la commande suivante:

```
strmqm winnt
```

où *winnt* est le nom donné au gestionnaire de files d'attente lors de sa création.

2. Configurez les canaux pour l'exemple de configuration.

Pour plus d'informations sur les paramètres utilisés dans les exemples suivants, voir [«Paramètres de configuration de canal pour Windows»](#), à la page 46. Dans chaque cas, l'exemple montre la commande MQSC. Démarrez **runmqsc** à partir d'une ligne de commande Linux et entrez chaque commande à tour de rôle, ou générez les commandes dans un fichier de commandes. Ces exemples permettent de connecter IBM MQ for Windows et IBM MQ for AIX. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs appropriées des tables de [«Paramètres de configuration de canal pour Windows»](#), à la page 46 à la place des valeurs de IBM MQ for AIX.

- a) Définissez le canal émetteur comme illustré dans les exemples suivants:

- Utilisation de SNA

```
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

```

- A l'aide de TCP

```

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) +                             F
  replace

```

b) Définissez le canal récepteur comme illustré dans les exemples suivants:

- Utilisation de SNA:

```

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

```

- Utilisation de TCP:

```

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

```

Que faire ensuite

démarrage automatique

IBM MQ for Windows permet d'automatiser le démarrage d'un gestionnaire de files d'attente et de son initiateur de canal, de ses canaux, de ses programmes d'écoute et de ses serveurs de commandes.

Utilisez le composant logiciel enfichable IBM MQ Services pour définir les services du gestionnaire de files d'attente. Une fois que vous avez terminé avec succès le test de votre configuration des communications, définissez les services appropriés sur **automatique** dans le composant logiciel enfichable. Ce fichier peut être lu par le service IBM MQ fourni lorsque le système est démarré.

Pour plus d'informations, voir [Administration de IBM MQ](#).

Exécution de canaux en tant que processus ou unités d'exécution

IBM MQ for Windows offre la possibilité d'exécuter des canaux d'envoi en tant que processus Windows ou unités d'exécution Windows. Ce paramètre est spécifié dans le paramètre MCATYPE de la définition de canal émetteur.

La plupart des installations exécutent leurs canaux d'envoi en tant qu'unités d'exécution, car la mémoire virtuelle et réelle requise pour prendre en charge de nombreuses connexions de canal simultanées est

réduite. Toutefois, une connexion NetBIOS nécessite un processus distinct pour l'agent Message Channel d'envoi.

Windows Paramètres de configuration de canal pour Windows

Paramètres requis pour configurer les canaux pour l'exemple de configuration sur Windows.

L'étape «2», à la page 44 de «Configuration des canaux sur Windows», à la page 43 décrit la configuration à effectuer sur le gestionnaire de files d'attente Linux pour implémenter le canal décrit dans «Exemple: configuration de la communication multiplateforme pour IBM MQ», à la page 5. Les exemples dans «Configuration des canaux sur Windows», à la page 43 permettent de connecter IBM MQ for Windows et IBM MQ for AIX. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez les valeurs de la table appropriée à la place des valeurs de Windows.

Remarque : Les mots en **gras** sont des valeurs suggérées et reflètent les noms des objets IBM MQ utilisés dans ces exemples. Vous pouvez les modifier dans l'installation de votre produit mais, si vous le faites, veillez à utiliser vos propres valeurs lors de l'utilisation des exemples de cette section.

Dans chaque cas, la commande MQSC est affichée. Démarrez **runmqsc** à partir d'une invite de commande et entrez chaque commande à tour de rôle, ou générez les commandes dans un fichier de commandes.

Des exemples sont fournis pour la connexion de IBM MQ for Windows et de IBM MQ for AIX. Pour vous connecter à IBM MQ sur une autre plateforme, utilisez l'ensemble de valeurs approprié de la table à la place de celles de Windows.

Définition du noeud local

Tableau 25. Exemples de configuration pour la définition du noeud local

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
A	Nom du gest. de files		winnt
B	Nom de la file d'attente locale		WINNT.LOCALQ

Connexion à IBM MQ sur AIX

AIX

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour AIX», à la page 13, comme indiqué.

Tableau 26. Exemples de configuration pour la connexion à IBM MQ sous AIX

	Parameter Name	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	AIX
D	Nom de la file d'attente distante		AIX:REMOTEQ
E	Nom de la file d'attente sur le système distant	B	AIX.LOCALQ
F	Nom de la file d'attente de transmission		AIX
G	Nom du canal émetteur (SNA)		WINNT.AIX.SNA
H	Nom du canal émetteur (TCP)		WINNT.AIX.TCP
I	Nom de canal récepteur (SNA)	G	AIX.WINNT.SNA
J	Nom du canal récepteur (TCP)	H	AIX.WINNT.TCP

Connexion à IBM MQ sur IBM i

IBM i

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Paramètres de configuration de canal pour IBM i», à la page 31, comme indiqué.

Tableau 27. Exemples de configuration pour la connexion à IBM MQ sous IBM i

division d'identification	Parameter Name	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	AS400
D	Nom de la file d'attente distante		AS400.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	AS400.LOCALQ
F	Nom de la file d'attente de transmission		AS400
G	Nom du canal émetteur (SNA)		WINNT.AS400.SNA
H	Nom du canal émetteur (TCP)		WINNT.AS400.TCP
I	Nom de canal récepteur (SNA)	G	AS400.WINNT.SNA
J	Nom du canal récepteur (TCP)	H	AS400.WINNT.TCP

connexion à IBM MQ for z/OS

z/OS

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Channel configuration parameters for z/OS», à la page 50, comme indiqué.

Tableau 28. Exemples de configuration pour la connexion à IBM MQ for z/OS

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	système MVS
D	Nom de la file d'attente distante		MVS.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	MVS.LOCALQ
F	Nom de la file d'attente de transmission		système MVS
G	Nom du canal émetteur (SNA)		WINNT.MVS.SNA
H	Nom du canal émetteur (TCP)		WINNT.MVS.TCP
I	Nom de canal récepteur (SNA)	G	MVS.WINNT.SNA
J	Nom du canal récepteur (TCP)	H	MVS.WINNT.TCP

Connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente

z/OS

Les valeurs de cette section du tableau doivent correspondre à celles utilisées dans «Shared channel configuration parameters», à la page 59, comme indiqué.

Tableau 29. Exemples de configuration pour la connexion à IBM MQ for z/OS à l'aide de groupes de partage de files d'attente

division d'identification	Nom du paramètre :	Référence	Exemple utilisé
C	Nom du gestionnaire de files d'attente éloignées	A	QSG
D	Nom de la file d'attente distante		QSG.REMOTEQ
E	Nom de la file d'attente sur le système distant	B	QSG.SHAREDQ
F	Nom de la file d'attente de transmission		QSG
G	Nom du canal émetteur (SNA)		WINNT.QSG.SNA
H	Nom du canal émetteur (TCP)		WINNT.QSG.TCP
I	Nom de canal récepteur (SNA)	G	QSG.WINNT.SNA
J	Nom du canal récepteur (TCP)	H	QSG.WINNT.TCP

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 [“Exemple: configuration de la communication multiplateforme pour IBM MQ”](#) on page 5 and [“Comment utiliser les exemples de communication multiplateforme”](#) 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
-  AIX
-  Linux
-  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 49 for details on how you configure the channels.

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 50. 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 50 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           F
  Usage      : X (XmitQ)
```

```

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      F
  Usage : X (XmitQ)

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 49 of “Configuring the channels on IBM MQ for z/OS” on page 49 describes the configuration to be performed on the z/OS queue manager to implement the channel described in “Exemple: configuration de la communication multiplateforme pour IBM MQ” on page 5. The examples in “Configuring the channels on IBM MQ for z/OS” on page 49 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

Windows

The values in this section of the table must match the values used in “[Paramètres de configuration de canal pour Windows](#)” on page 46, 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

AIX

The values in this section of the table must match the values used in “[Paramètres de configuration de canal pour AIX](#)” on page 13, as indicated.

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

ID	Parameter Name	Reference	Example Used
<p>AIX AIX Connection to IBM MQ for AIX</p>			
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

Linux

The values in this section of the table must match the values used in “Paramètres de configuration de canal pour Linux” on page 39, 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

IBM i

The values in this section of the table must match the values used in “Paramètres de configuration de canal pour IBM i” on page 31, 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
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 48.

For background information about this example and how to use it, see [“Exemple: configuration de la communication multiplateforme pour IBM MQ”](#) on page 5 and [“Comment utiliser les exemples de communication multiplateforme”](#) 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 53.
 - Establish a TCP connection using Sysplex Distributor as described in [“Establishing a TCP connection using Sysplex Distributor”](#) on page 57.
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 57 for details of this process.

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(mslu1)
      NOSCHED
      TPDATA(csq.appctp)
      GRNAME(mvsgt)
```

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 ATBSDLFMU 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
```

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 57.

z/OS 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 53, 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 56.

Definition for 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
11	Symbolic destination		G1
12	Symbolic destination for generic resource name		G2

Connection to IBM MQ on Windows

Windows

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 57.

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 59](#). 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 59](#) 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
  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 : QSG.WINNT.TCP
  Transport type : T (TCP)
  Transmission queue name : WINNT
  Connection name : winnt.tcpip.hostname
  Disposition : GROUP
  H
  F
```

- 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 58 of “[Configuring shared channels on IBM MQ for z/OS](#)” on page 57 describes the configuration to be performed on the z/OS queue manager to implement the channel described in “[Exemple: configuration de la communication multiplateforme pour IBM MQ](#)” on page 5. The examples in “[Configuring shared channels on IBM MQ for z/OS](#)” on page 57 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

The values in this section of the table must match the values used in “[Paramètres de configuration de canal pour Windows](#)” on page 46, 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

ID	Parameter name	Reference	Example used
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



The values in this section of the table must match the values used in [“Paramètres de configuration de canal pour AIX”](#) on page 13, as indicated.

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 61.
2. Set up and run Configuration 2.
For more information, see [“Setting up and running configuration 2”](#) on page 63.
3. Set up and run Configuration 3.
For more information, see [“Setting up and running configuration 3”](#) on page 66.

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).

z/OS 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.

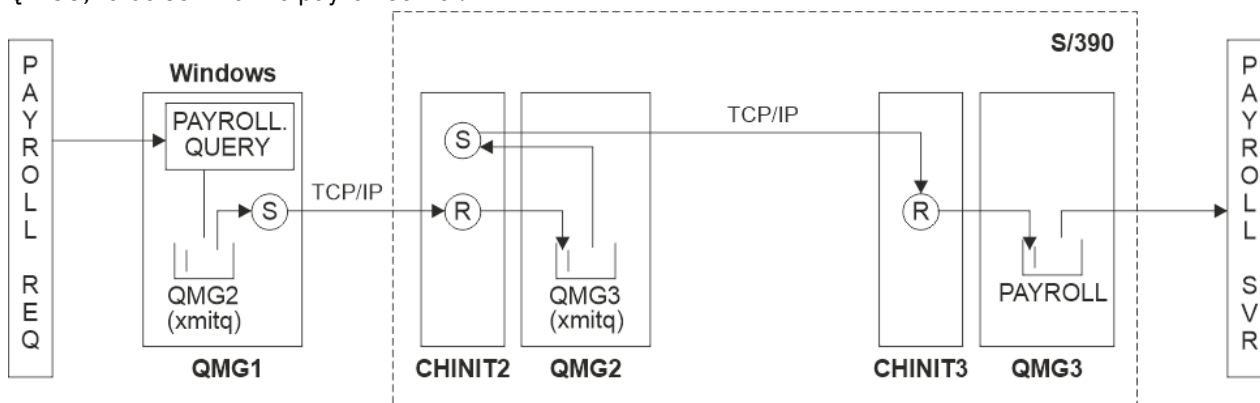


Figure 2. Configuration 1: z/OS using intra-group queuing

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.

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.

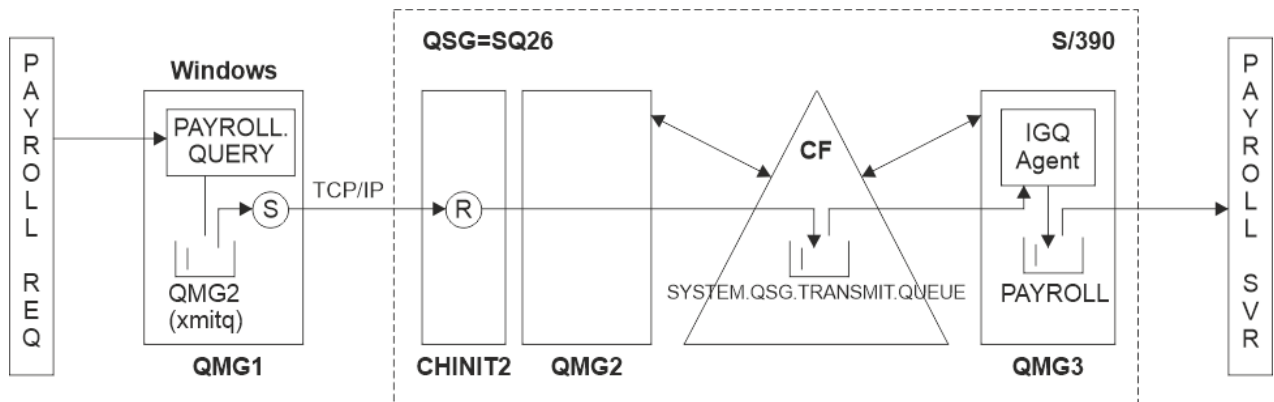


Figure 3. Configuration 2

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:

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 the sender channel 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.

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.

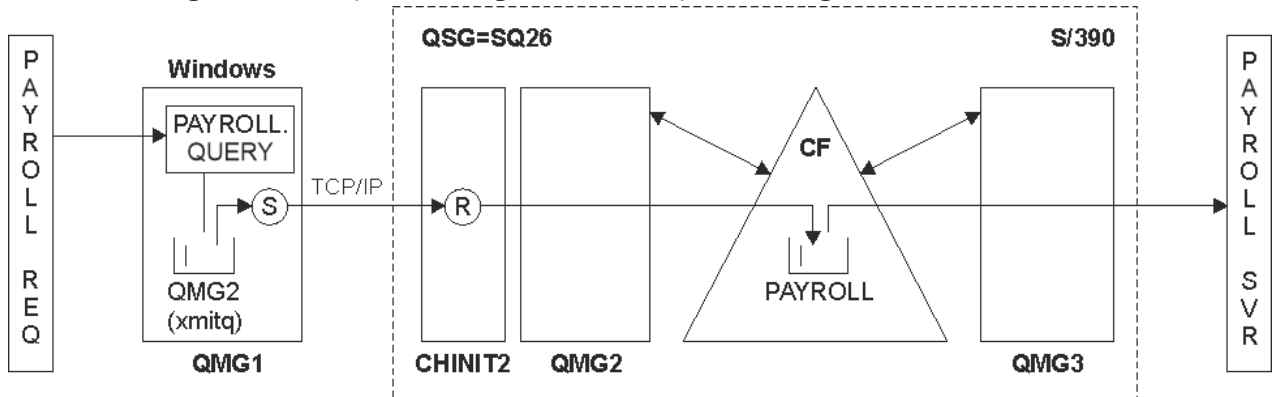


Figure 4. Configuration 3

The flow of operations is:

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 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 171.

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 **Droits d'accès au système de fichiers IBM MQ appliqués à /var/mqm**

Les informations suivantes décrivent la sécurité appliquée aux fichiers et aux répertoires sous /var/mqm/ et la raison pour laquelle les droits du système de fichiers sont définis tels qu'ils sont. Afin de garantir le bon fonctionnement de IBM MQ, vous ne devez pas modifier les droits d'accès au système de fichiers définis par IBM MQ

commande **crtmqdir**

Si votre entreprise a modifié l'un des droits d'accès aux fichiers /var/mqm, pour quelque raison que ce soit, vous pouvez mettre à jour les droits d'accès ou ajouter des répertoires à l'aide de la commande **crtmqdir**

Sécurité du système de fichiers IBM MQ sous AIX, Linux et IBM i

Les fichiers du répertoire de données IBM MQ (/var/mqm) sont utilisés pour stocker:

- IBM MQ données de configuration
- Données d'application (objets IBM MQ et données contenues dans les messages IBM MQ)
- Informations de contrôle d'exécution
- Informations de surveillance (messages et fichiers FFST)

L'accès à ces données est contrôlé à l'aide des droits du système de fichiers, certaines des données étant accessibles à tous les utilisateurs, tandis que les autres données sont limitées uniquement aux membres du IBM MQ groupe d'administrateurs 'mqm' (ou QMQM sous IBM i).

L'accès est accordé dans les trois catégories suivantes:

Groupe **mqm** uniquement

Les fichiers et répertoires de cette catégorie ne sont accessibles qu'aux administrateurs IBM MQ (membres du groupe 'mqm') et aux processus du gestionnaire de files d'attente IBM MQ.

Les droits d'accès aux fichiers pour ces fichiers et répertoires sont les suivants:

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

Voici un exemple de fichiers et de répertoires dans cette catégorie:

```
/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/
...
```

Accès en lecture à tous les utilisateurs-accès en lecture et en écriture aux membres du groupe mqm

Les fichiers et répertoires de cette catégorie peuvent être lus par tous les utilisateurs, mais seuls les membres du groupe 'mqm' peuvent modifier ces fichiers et manipuler ces répertoires.

Les droits d'accès aux fichiers pour ces fichiers et répertoires sont les suivants:

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

Voici un exemple de fichiers et de répertoires dans cette catégorie:

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



Attention : Vous devez uniquement définir des droits d'exécution sur les fichiers exécutables et les scripts. Par exemple, sous Linux, lorsque la commande **crtmqm** est exécutée, les droits d'accès aux fichiers suivants sont définis:

```
-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
```

Accès en lecture et en écriture pour tous les utilisateurs

Fichiers disposant d'un accès en lecture et en écriture pour tous les utilisateurs

IBM MQ n'a pas de fichiers *standard* disposant de droits d'accès en écriture pour tous les utilisateurs (777). Cependant, il existe un certain nombre de fichiers *spéciaux* qui semblent disposer de droits d'accès en écriture pour tous les utilisateurs.

Ces fichiers spéciaux ne présentent aucun risque pour la sécurité. Bien que les droits soient affichés sous la forme 777, il ne s'agit pas de fichiers *standard* et vous ne pouvez pas y écrire directement.

Ces fichiers spéciaux sont les suivants:

Liens symboliques

Les liens symboliques sont identifiés par le caractère 'l' au début de leurs droits. Les droits d'accès au lien symbolique n'ont aucun effet sur les personnes qui peuvent accéder au fichier cible, car l'accès à la commande est contrôlé par les droits d'accès sur la cible du lien symbolique.

Sur la plupart des systèmes AIX and Linux, il n'est pas possible de modifier les droits d'accès sur les liens symboliques, de sorte qu'ils apparaissent toujours sous la forme `lrwxrwxrwx`.

Fichiers de socket

Les fichiers de socket sont des fichiers spéciaux créés par le système d'exploitation suite à la création d'un socket de domaine UNIX par un processus. Ces fichiers peuvent être identifiés par le 's' au début des droits d'accès aux fichiers, c'est-à-dire `srwxrwxrwx`.

Les droits d'accès au fichier n'accordent pas l'accès au fichier lui-même, mais définissent qui peut se connecter au socket de domaine UNIX.

IBM MQ utilise un certain nombre de ces fichiers de socket et les droits sont toujours définis en fonction de la personne autorisée à communiquer avec le socket.

Les répertoires suivants contiennent des fichiers de socket disposant de droits de lecture / écriture pour tous les utilisateurs (srxwxrwxrwx).

IBM MQ 8.0:

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

Fichiers de socket utilisés par les applications qui se connectent à IBM MQ à l'aide de liaisons isolées.

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

Répertoires disposant d'un accès en lecture et en écriture pour tous les utilisateurs

Parfois, les applications IBM MQ doivent créer des fichiers dans le répertoire de données IBM MQ . Pour garantir que les applications puissent créer des fichiers lorsqu'elles sont requises, un certain nombre de répertoires bénéficient d'un accès en écriture à l'échelle mondiale, ce qui signifie que tout utilisateur du système peut créer des fichiers dans ce répertoire.

A l'exception des fichiers journaux d'erreurs, qui peuvent être écrits par n'importe quel membre du groupe 'mqm', tous les fichiers créés dans ces répertoires sont créés avec des droits d'accès restreints qui n'autorisent que l'accès en écriture du créateur de fichier. Cela permet à l'administrateur système de suivre l'ID utilisateur de toutes les données écrites dans les fichiers de ces répertoires.

/var/mqm/errors/

Ce répertoire contient les fichiers journaux des erreurs système et les fichiers FFST. Les droits d'accès à ce répertoire sont 'dwxrwsrwt', ce qui signifie que tous les utilisateurs du système peuvent créer des fichiers dans ce répertoire.

Le bit SetGroupId 's'indique que tous les fichiers créés dans ce répertoire ont la propriété de groupe 'mqm'.

Le bit de rappel 't' n'est pas défini par défaut sur ce répertoire, mais un administrateur IBM MQ peut le définir explicitement pour permettre aux utilisateurs de supprimer uniquement les fichiers qu'ils créent.

Remarque :  Cette fonction n'est pas disponible sous IBM i.

AMQERRO*.LOG

Ces fichiers journaux d'erreurs ne peuvent être écrits que directement par les membres du group , mais tout utilisateur peut lire les messages écrits dans ces fichiers (droit: -rw-rw-r--).

AMQnnnnn.*.FDC

Ces fichiers contiennent des informations FFST écrites lorsqu'une erreur se produit dans le gestionnaire de files d'attente ou dans une application écrite par un utilisateur. Ces fichiers sont créés avec les droits -rw-r-----.


/var/mqm/trace/

Les fichiers de trace sont écrits dans ce répertoire lorsque la fonction de trace IBM MQ est activée. La trace IBM MQ est écrite par tous les processus associés à un gestionnaire de files d'attente pour lequel la trace est activée.

Les droits d'accès à ce répertoire sont 'dwxrwsrwt', ce qui signifie que tous les utilisateurs du système peuvent créer des fichiers dans ce répertoire.

Le bit SetGroupId 's'indique que tous les fichiers créés dans ce répertoire ont la propriété de groupe 'mqm'.

Le bit de rappel 't' n'est pas défini par défaut sur ce répertoire, mais un administrateur IBM MQ peut le définir explicitement pour permettre aux utilisateurs de supprimer uniquement les fichiers qu'ils créent.

Remarque :  Cette fonction n'est pas disponible sous IBM i.

AMQnnnnn. *.TRC

Ces fichiers contiennent les données de trace écrites par chaque processus qui est en cours de traçage et sont créés avec des droits d'accès -`rw-r-----`

Les droits d'accès à ce répertoire sont `drwxrwsrwt` et les droits d'accès aux fichiers de socket créés dans ce répertoire sont `srwx-----`.

IBM MQ 8.0:

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


Ce répertoire est utilisé par les applications qui se connectent au gestionnaire de files d'attente IBM MQ à l'aide de liaisons *isolées*. Lors du traitement de la connexion, un fichier socket est créé par l'application de connexion dans ce répertoire. Le fichier de socket est supprimé une fois la connexion au gestionnaire de files d'attente établie.

Les droits d'accès à ce répertoire sont `drwxrwsrwt` et les droits d'accès aux fichiers de socket créés dans ce répertoire sont `srwx-----`.

Le bit `SetGroupId` 's' de ce répertoire garantit que tous les fichiers créés dans ce répertoire ont la propriété de groupe 'mqm'.

Sur toutes les plateformes sauf IBM i, ce répertoire possède également le bit de rappel 't' qui empêche un utilisateur de supprimer tous les fichiers sauf ceux dont il est propriétaire. Cela empêche un utilisateur non autorisé de supprimer des fichiers qu'il ne possède pas.

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

 Pour les processus qui se connectent à IBM MQ à l'aide de liaisons *partagées*, des sockets de domaine UNIX peuvent être utilisés pour la synchronisation entre l'application et le gestionnaire de files d'attente. Lorsque des sockets de domaine UNIX sont utilisés, le fichier de socket associé est créé dans ces répertoires.

Les droits d'accès à ces répertoires sont `drwxrwsrwt` et les droits d'accès aux fichiers de socket créés dans ces répertoires sont `srwxrwxrwx`.

Le bit `SetGroupId` 's' sur ces répertoires garantit que tous les fichiers créés dans ces répertoires ont la propriété de groupe 'mqm'.

Sur toutes les plateformes sauf IBM i, le bit de rappel 't' est également défini pour ces répertoires, ce qui empêche un utilisateur de supprimer tous les fichiers à l'exception de ceux dont il est propriétaire. Cela empêche un utilisateur non autorisé de supprimer des fichiers qu'il ne possède pas.

ACCUEIL

Un répertoire `${HOME}/.mqm` est créé lors de l'utilisation d'une version non enregistrée ou non installée de IBM MQ, telle que le client redistribuable.

Le répertoire est créé pour qu'IBM MQ puisse accéder de manière fiable à ses fichiers de socket via un chemin dont la longueur est conforme au paramètre **sun_path**. Si IBM MQ ne parvient pas à écrire dans le répertoire HOME, vous recevez un message d'erreur.

Utilisation des ressources System V IPC par IBM MQ

IBM MQ utilise la mémoire partagée System V et les sémaphores pour la communication inter-processus. Ces ressources sont regroupées en fonction de la manière dont elles sont utilisées avec chaque groupe disposant des droits de propriété et d'accès appropriés.

Pour vérifier quelles ressources System V IPC d'un système appartiennent à IBM MQ, vous pouvez:

- Vérifiez la propriété.

L'utilisateur propriétaire des ressources IBM MQ System V IPC est toujours l'utilisateur 'mqm' sur les plateformes AIX and Linux . Sous IBM i , l'utilisateur propriétaire est 'QMQM'.

- IBM MQ 8.0 et versions ultérieures, utilisez l'utilitaire amqspdbg .

L'utilitaire amqspdbg fourni avec IBM MQ peut être utilisé pour afficher la mémoire partagée et les ID sémaphore d'un gestionnaire de files d'attente donné.

Vous devez exécuter la commande une fois pour le groupe 'system' des ressources System V créées par IBM MQ

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

puis quatre fois pour chaque gestionnaire de files d'attente sur le système afin d'obtenir la liste complète des ressources System V utilisées par IBM MQ. Supposons que le nom du gestionnaire de files d'attente est QMGR1 dans les exemples suivants:

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

Les droits d'accès sur les ressources System V créées par IBM MQ sont définis pour n'accorder que le niveau d'accès correct aux utilisateurs autorisés. Un certain nombre de ressources System V IPC créées par IBM MQ sont accessibles à tous les utilisateurs sur la machine et disposent des droits d'accès -rwx-rwx-rwx-.

Le paramètre **-g ApplicationGroup** de la commande `crtmqm` peut être utilisé pour limiter l'accès à un gestionnaire de files d'attente à l'appartenance à un groupe de systèmes d'exploitation spécifique. L'utilisation de cette fonctionnalité de groupe restreint limite davantage les droits accordés sur les ressources IPC System V.

Linux AIX Droits d'accès aux fichiers IBM MQ dans /opt/mqm avec setuid pour mqm

Les informations suivantes concernent le cas où votre équipe de sécurité a marqué certains des fichiers IBM MQ exécutables dans l'arborescence de répertoires `$MQ_INSTALLATION_PATH`, en violation des règles de sécurité locales. L'emplacement par défaut dans AIX est `/usr/mqm` et pour les autres systèmes d'exploitation UNIX , il s'agit de `/opt/mqm`. Si vous avez installé IBM MQ dans un répertoire autre que celui par défaut, tel que `/opt/mqm90`, ou si vous disposez de plusieurs installations, les détails de cette rubrique s'appliquent toujours.

Cause du problème

Votre équipe de sécurité a identifié les domaines de préoccupation suivants sous `$MQ_INSTALLATION_PATH`:

1. Les fichiers du répertoire `/opt/mqm/bin` sont `setuid` pour le propriétaire de l'arborescence de répertoires où ils résident. Exemple :

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

2. Pratiquement tous les répertoires et fichiers appartiennent à "mqm:mqm", à l'exception des suivants, qui appartiennent à root:

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


Ce sous-répertoire doit appartenir à root, car il s'agit des fichiers exécutables qui interagissent avec le système d'exploitation lorsque l'utilisateur d'un client IBM MQ spécifie un mot de passe, et ce mot de passe est transmis par le gestionnaire de files d'attente IBM MQ au système d'exploitation pour confirmer si le mot de passe est valide ou non.

3. L'utilisateur ne possède pas de fichiers dans le répertoire `/opt/mqm/lib/iconv` (ce répertoire n'existe pas sur AIX). Exemple :

```
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. Répertoire de maintenance du groupe de correctifs sur les systèmes Linux basés sur RPM. Lorsque des groupes de correctifs sont installés, les fichiers existants sont sauvegardés sous ce répertoire dans une structure similaire à celle illustrée dans l'exemple suivant, à ceci près que dans cet exemple, V.R représente le numéro de version et d'édition de IBM MQ et les sous-répertoires qui apparaissent dépendent des groupes de correctifs qui ont été installés:

```
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
...
```

Résolution du problème

L'une des préoccupations sur les systèmes UNIX concernant les programmes setuid était que la sécurité du système pouvait être compromise en manipulant des variables d'environnement telles que LD* (LD_LIBRARY_PATH, LIBPATH sur AIX, etc.). Cela n'est plus une préoccupation, car les différents systèmes d'exploitation UNIX ignorent désormais ces variables d'environnement LD* lors du chargement des programmes setuid.

1. Pourquoi certains des programmes IBM MQ sont mqm-setuid ou mqm-setgid.

Dans IBM MQ, l'ID utilisateur "mqm" et tout ID faisant partie du groupe "mqm" sont les administrateurs de IBM MQ .

Les ressources du gestionnaire de files d'attente IBM MQ sont protégées par l'authentification auprès de cet utilisateur. Etant donné que les processus de gestionnaire de files d'attente utilisent et modifient ces ressources de gestionnaire de files d'attente, les processus de gestionnaire de files d'attente nécessitent des droits d'accès "mqm" pour accéder aux ressources. Par conséquent, les processus de prise en charge du gestionnaire de files d'attente IBM MQ sont conçus pour s'exécuter avec l'ID utilisateur effectif "mqm".

Pour aider les utilisateurs non administrateurs à accéder aux objets IBM MQ , IBM MQ fournit une fonction OAM (Object Authority Manager), qui permet d'accorder et de révoquer des droits en fonction des besoins de l'application exécutée par l'utilisateur non administrateur.

Avec la possibilité d'accorder différents niveaux d'authentification pour les utilisateurs et le fait que les programmes **setuid** et **setgid** ignorent les variables LD* , les fichiers binaires et de bibliothèque IBM MQ ne compromettent en aucune manière la sécurité de votre système.

2. Il n'est pas possible de modifier les droits d'accès pour satisfaire les règles de sécurité de votre entreprise sans compromettre la fonctionnalité IBM MQ .

Vous ne devez pas modifier les droits et les propriétés des fichiers binaires et des bibliothèques IBM MQ . La fonctionnalité IBM MQ peut pâtir de ce type de changement, de sorte que les processus de gestionnaire de files d'attente risquent de ne pas pouvoir accéder à certaines des ressources.

Notez que les droits et les propriétés ne représentent aucune menace pour la sécurité du système.

Les disques durs / disques Linux sur lesquels IBM MQ est installé ou sur lesquels se trouvent les données IBM MQ ne doivent pas être montés avec l'option `nosuid` . Cette configuration peut empêcher la fonctionnalité IBM MQ .

Pour plus d'informations, voir [«Droits d'accès au système de fichiers IBM MQ appliqués à /var/mqm»](#), à la page 68.

Concepts associés

[Système de fichiers](#)

Windows Droits d'accès au système de fichiers IBM MQ sur Windows

Les informations suivantes décrivent la sécurité appliquée aux fichiers et aux répertoires sous Windows. Afin de garantir le bon fonctionnement de IBM MQ, vous ne devez pas modifier les droits d'accès au système de fichiers définis par IBM MQ.

Répertoire de données

Remarque : Les droits définis à la racine de ce répertoire sont hérités vers le bas dans l'ensemble de la structure de répertoire.

Les répertoires sous le répertoire de données (DATADIR) sont définis avec les droits suivants, à l'exception des exceptions détaillées dans le texte suivant.

Administrateurs

Contrôle total

groupe mqm

Contrôle total

SYSTEME

Contrôle total

Tous

Lecture et exécution

Voici les exceptions :

DATADIR \erreurs

Tout le monde contrôle

DATADIR \trace

Tout le monde contrôle

DATADIR \journal

Administrateurs

Contrôle total

groupe mqm

Contrôle total

SYSTEME

Contrôle total

Tous

Lisez

DATADIR \log \ < nom_gest_files_attente > \active

Administrateurs

Contrôle total

groupe mqm

Contrôle total

SYSTEME

Contrôle total

Aucun accès accordé à Tout le monde.

Fichiers journaux des erreurs AMQERR01.LOG, et ainsi de suite, n'hérite pas de leurs paramètres de sécurité de leur répertoire, mais est défini sur Everyone: Full Control.

Versions antérieures du produit

Dans les versions du produit antérieures à IBM MQ 8.0, le programme par défaut et les répertoires de données par défaut étaient co-localisés.

Dans toute installation qui a été installée à l'origine avant IBM MQ 8.0. et qui a été installé dans les emplacements par défaut, puis mis à niveau à partir de là, les répertoires de données et de programme restent colocalisé (dans C:\Program Files\IBM\WebSphere MQ).

Dans le cas de répertoires de données et de programmes co-localisés, les informations précédentes s'appliquent uniquement aux répertoires qui appartiennent au répertoire de données et non à ceux qui font partie du répertoire de programmes.

Restrictions de dénomination pour les files d'attente

La longueur des noms de file d'attente est soumise à des restrictions. Certains noms de file d'attente sont réservés aux files d'attente définies par le gestionnaire de files d'attente.

Restrictions sur la longueur des noms

Les noms des files d'attente peuvent comporter jusqu'à 48 caractères.

Noms de file d'attente réservés

Les noms commençant par "SYSTEM." sont réservés aux files d'attente définies par le gestionnaire de files d'attente. Vous pouvez utiliser les commandes **ALTER** ou **DEFINE REPLACE** pour modifier ces définitions de file d'attente en fonction de votre installation. Les noms suivants sont définis pour IBM MQ:

Nom de la file d'attente	Description
SYSTEM.ADMIN.ACTIVITY.QUEUE	File d'attente pour les rapports d'activité
SYSTEM.ADMIN.CHANNEL.EVENT	File d'attente des événements de canal
SYSTEM.ADMIN.COMMAND.EVENT	File d'attente des événements de commande
SYSTEM.ADMIN.COMMAND.QUEUE	File d'attente dans laquelle les messages de commande PCF sont envoyés
SYSTEM.ADMIN.CONFIG.EVENT	File d'attente des événements de configuration
SYSTEM.ADMIN.PERFM.EVENT	File d'attente des événements de performances
SYSTEM.ADMIN.PUBSUB.EVENT	File d'attente d'événements liés à la publication / l'abonnement du système
SYSTEM.ADMIN.QMGR.EVENT	File d'attente des événements du gestionnaire de files d'attente
SYSTEM.ADMIN.TRACE.ROUTE.QUEUE	File d'attente des messages de réponse de trace-route
SYSTEM.AUTH.DATA.QUEUE	File d'attente contenant les listes de contrôle d'accès pour le gestionnaire de files d'attente. (Pas pour z/OS)
SYSTEM.CHANNEL.INITQ	File d'attente d'initialisation pour les canaux
SYSTEM.CHANNEL.SYNCQ	File d'attente contenant les données de synchronisation pour les canaux
SYSTEM.CHLAUTH.DATA.QUEUE	File d'attente de données d'authentification de canal IBM MQ

Tableau 41. Noms et descriptions des files d'attente réservées (suite)

Nom de la file d'attente	Description
SYSTEM.CICS.INITIATION.QUEUE	File d'attente utilisée pour le déclenchement (non pour z/OS)
SYSTEM.CLUSTER.COMMAND.QUEUE	File d'attente utilisée pour communiquer les modifications de référentiel entre les gestionnaires de files d'attente
SYSTEM.CLUSTER.HISTORY.QUEUE	La file d'attente est utilisée pour stocker l'historique des informations d'état de cluster à des fins de service.
SYSTEM.CLUSTER.REPOSITORY.QUEUE	File d'attente utilisée pour stocker les informations sur le référentiel
SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE	File d'attente utilisée pour créer des files d'attente de transmission individuelles pour chaque canal émetteur de cluster.
SYSTEM.CLUSTER.TRANSMIT.QUEUE	File d'attente de transmission pour toutes les destinations gérées par le support de cluster
SYSTEM.COMMAND.INPUT	File d'attente dans laquelle les messages de commande sont envoyés sur z/OS
SYSTEM.COMMAND.REPLY.MODEL	Définition de file d'attente modèle pour les réponses aux commandes (pour z/OS)
SYSTEM.DEAD.LETTER.QUEUE	File d'attente de rebut (pas pour z/OS)
SYSTEM.DEFAULT.ALIAS.QUEUE	Définition de file d'attente alias par défaut
SYSTEM.DEFAULT.INITIATION.QUEUE	File d'attente utilisée pour déclencher un processus spécifié (pas pour z/OS)
SYSTEM.DEFAULT.LOCAL.QUEUE	Définition de file d'attente locale par défaut
SYSTEM.DEFAULT.MODEL.QUEUE	Définition de file d'attente modèle par défaut
SYSTEM.DEFAULT.REMOTE.QUEUE	Définition de file d'attente éloignée par défaut
SYSTEM.DURABLE.SUBSCRIBER.QUEUE	File d'attente locale utilisée pour stocker une copie persistante des abonnements durables dans le gestionnaire de files d'attente
SYSTEM.HIERARCHY.STATE	File d'attente utilisée pour stocker des informations sur l'état des relations entre gestionnaires de files d'attente dans une hiérarchie de publication / abonnement
SYSTEM.JMS.TEMPQ.MODEL	Modèle pour les files d'attente temporaires JMS
SYSTEM.INTERNAL.REPLY.QUEUE	File d'attente de réponses interne IBM MQ (non pour z/OS)
SYSTEM.INTER.QMGR.CONTROL	File d'attente utilisée dans une hiérarchie de publication / abonnement pour recevoir des demandes d'un gestionnaire de files d'attente éloignées afin de créer un abonnement de proxy
SYSTEM.INTER.QMGR.PUBS	File d'attente utilisée dans une hiérarchie de publication / abonnement pour recevoir des publications d'un gestionnaire de files d'attente éloignées

Tableau 41. Noms et descriptions des files d'attente réservées (suite)

Nom de la file d'attente	Description
SYSTEM . INTER . QMGR . FANREQ	File d'attente utilisée dans une hiérarchie de publication / abonnement pour traiter les demandes de création d'un abonnement de proxy sur un gestionnaire de files d'attente éloignées
SYSTEM . MQEXPLORER . REPLY . MODEL	Définition de file d'attente modèle pour les réponses pour IBM MQ Explorer
SYSTEM . MQSC . REPLY . QUEUE	Définition de file d'attente modèle pour les réponses de commande MQSC (non pour z/OS)
SYSTEM . QSG . CHANNEL . SYNCQ	File d'attente locale partagée utilisée pour le stockage des messages contenant les informations de synchronisation pour les canaux partagés (z/OS uniquement)
SYSTEM . QSG . TRANSMIT . QUEUE	File d'attente locale partagée utilisée par l'agent de mise en file d'attente intra-groupe lors de la transmission de messages entre les gestionnaires de files d'attente du même groupe de partage de files d'attente (z/OS uniquement)
SYSTEM . RETAINED . PUB . QUEUE	File d'attente locale utilisée pour conserver une copie de chaque publication conservée dans le gestionnaire de files d'attente.
SYSTEM . SELECTION . EVALUATION . QUEUE	File d'attente d'évaluation de sélection interne IBM MQ (non pour z/OS)
SYSTEM . SELECTION . VALIDATION . QUEUE	File d'attente de validation de sélection interne IBM MQ (non pour z/OS)

Restrictions de dénomination pour d'autres objets

La longueur des noms d'objet est soumise à des restrictions. Certains noms d'objet sont réservés aux objets définis par le gestionnaire de files d'attente.

Restrictions relatives à la longueur du nom

Les processus, les listes de noms, les clusters, les rubriques, les services et les objets d'informations d'authentification peuvent comporter jusqu'à 48 caractères.

Les noms des canaux peuvent comporter jusqu'à 20 caractères.

Les noms des classes de stockage peuvent comporter jusqu'à 8 caractères.

Les structures CF peuvent avoir des noms pouvant comporter jusqu'à 12 caractères.

Noms d'objet réservés

Noms commençant par SYSTEM. sont réservés à des objets définis par le gestionnaire de files d'attente. Vous pouvez utiliser les commandes **ALTER** ou **DEFINE REPLACE** pour modifier ces définitions d'objet en fonction de votre installation. Les noms suivants sont définis pour IBM MQ:

Tableau 42. Noms et descriptions d'objets réservés

Nom de l'objet	Description
SYSTEM.ADMIN.SVRCONN	Canal de connexion serveur utilisé pour l'administration à distance d'un gestionnaire de files d'attente
SYSTEM.AUTO.RECEIVER	Canal récepteur par défaut pour la définition automatique (systèmes AIX, Linux, and Windows uniquement)
SYSTEM.AUTO.SVRCONN	Canal de connexion serveur par défaut pour la définition automatique (Multiplatforms uniquement)
SYSTEM.BASE.TOPIC	Rubrique de base pour la résolution ASPARENT. Si un objet de rubrique d'administration particulier n'a pas d'objet de rubrique d'administration parent, tous les attributs ASPARENT sont hérités de cet objet
SYSTEM.DEF.CLNTCONN	Définition de canal de connexion client par défaut
SYSTEM.DEF.CLUSRCVR	Définition de canal récepteur de cluster par défaut
SYSTEM.DEF.CLUSSDR	Définition de canal émetteur de cluster par défaut
SYSTEM.DEF.RECEIVER	Définition de canal récepteur par défaut
SYSTEM.DEF.REQUESTER	Définition de canal demandeur par défaut
SYSTEM.DEF.SENDER	Définition de canal émetteur par défaut
SYSTEM.DEF.SERVER	Définition de canal serveur par défaut
SYSTEM.DEF.SVRCONN	Définition de canal de connexion serveur par défaut
SYSTEM.DEFAULT.AUTHINFO.CRLLDAP	Définition d'objet d'informations d'authentification par défaut pour la définition d'objets d'informations d'authentification de type CRLLDAP
SYSTEM.DEFAULT.AUTHINFO.OCSP	Définition d'objet d'informations d'authentification par défaut pour la définition d'objets d'informations d'authentification de type OCSP
SYSTEM.DEFAULT.LISTENER.LU62	Programme d'écoute SNA par défaut (Windows uniquement)
SYSTEME SYSTEM.DEFAULT.LISTENER.NETBIOS	Programme d'écoute NetBIOS par défaut (Windows uniquement)
SYSTEM.DEFAULT.LISTENER.SPX	Programme d'écoute SPX par défaut (Windows uniquement)
SYSTEM.DEFAULT.LISTENER.TCP	Programme d'écoute TCP/IP par défaut (Multiplatforms uniquement)
SYSTEM.DEFAULT.NAMELIST	Définition de liste de noms par défaut
SYSTEM.DEFAULT.PROCESS	Définition de processus par défaut
SYSTEM.DEFAULT.SERVICE	Service par défaut (Multiplatforms uniquement)
SYSTEM.DEFAULT.TOPIC	Définition de rubrique par défaut
SYSTEM.QPUBSUB.QUEUE.NAMELIST	Liste des files d'attente de l'interface de publication / abonnement en file d'attente à surveiller

Tableau 42. Noms et descriptions d'objets réservés (suite)

Nom de l'objet	Description
z/OS SYSTEMST	Définition de classe de stockage par défaut (z/OS uniquement)

Résolution de nom de file d'attente

Dans les grands réseaux, l'utilisation des gestionnaires de files d'attente présente un certain nombre d'avantages par rapport à d'autres formes de communication. Ces avantages découlent de la fonction de résolution de nom dans la gestion de files d'attente réparties, qui garantit que la résolution de nom de file d'attente est effectuée par les gestionnaires de files d'attente aux extrémités d'envoi et de réception d'un canal.

Les principaux avantages de cette approche sont les suivants:

- Les applications n'ont pas besoin de prendre des décisions de routage
- Les applications n'ont pas besoin de connaître la structure du réseau
- Les liens réseau sont créés par les administrateurs système
- La structure du réseau est contrôlée par les planificateurs de réseau
- Plusieurs canaux peuvent être utilisés entre les noeuds pour partitionner le trafic

La figure suivante illustre un exemple de résolution de nom de file d'attente. La figure montre deux machines dans un réseau, l'une exécutant une application d'insertion, l'autre exécutant une application d'extraction. Les applications communiquent entre elles via le canal IBM MQ, contrôlé par les MCM.

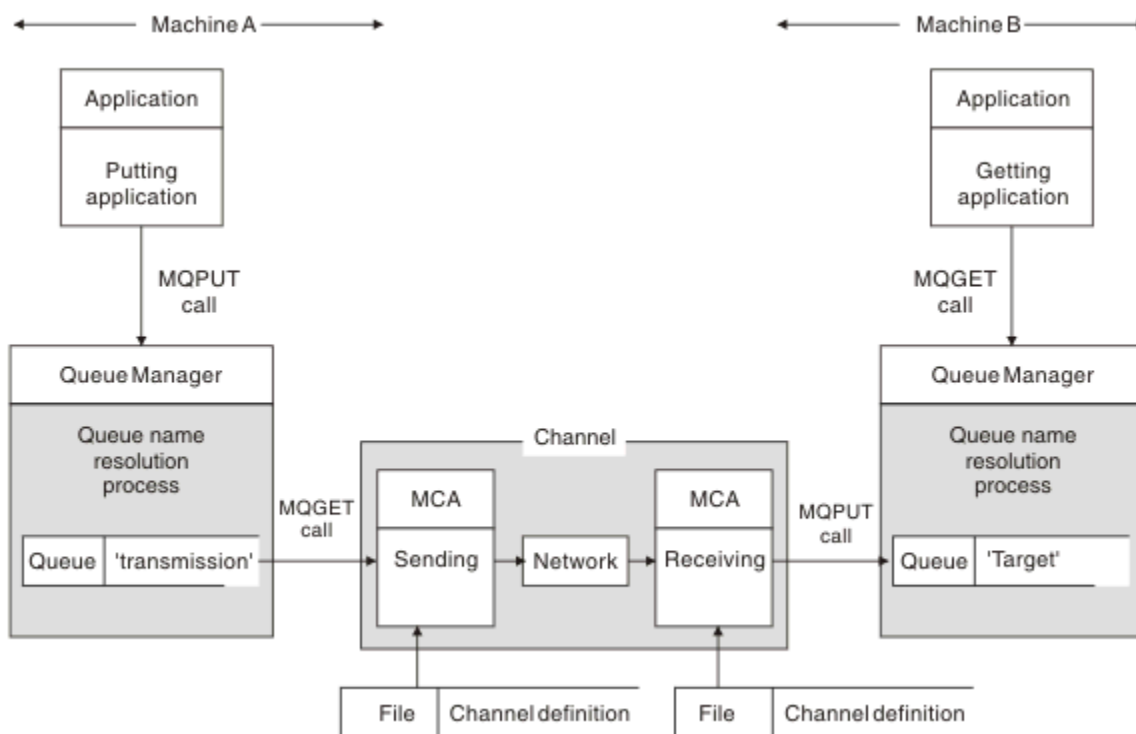


Figure 5. Résolution de nom

En référence à Figure 5, à la page 79, le mécanisme de base pour l'insertion de messages dans une file d'attente éloignée, en ce qui concerne l'application, est le même que pour l'insertion de messages dans une file d'attente locale:

- L'application qui place le message émet des appels MQOPEN et MQPUT pour placer des messages dans la file d'attente cible.

- L'application qui extrait les messages émet des appels MQOPEN et MQGET pour extraire les messages de la file d'attente cible.

Si les deux applications sont connectées au même gestionnaire de files d'attente, aucune communication entre les gestionnaires de files d'attente n'est requise et la file d'attente cible est décrite comme *locale* pour les deux applications.

Toutefois, si les applications sont connectées à des gestionnaires de files d'attente différents, deux agents MCA et leur connexion réseau associée sont impliqués dans le transfert, comme illustré dans la figure. Dans ce cas, la file d'attente cible est considérée comme une *file d'attente éloignée* de l'application d'insertion.

La séquence d'événements est la suivante :

1. L'application d'insertion émet des appels MQOPEN et MQPUT pour insérer des messages dans la file d'attente cible.
2. Lors de l'appel MQOPEN, la fonction de *résolution de nom* détecte que la file d'attente cible n'est pas locale et décide quelle file d'attente de transmission est appropriée. Ensuite, sur les appels MQPUT associés à l'appel MQOPEN, tous les messages sont placés dans cette file d'attente de transmission.
3. L'agent MCA émetteur extrait les messages de la file d'attente de transmission et les transmet à l'agent MCA récepteur sur l'ordinateur distant.
4. L'agent MCA récepteur place les messages dans la ou les files d'attente cible.
5. L'application d'extraction émet des appels MQOPEN et MQGET pour extraire les messages de la file d'attente cible.

Remarque : Seules les étapes 1 et 5 impliquent un code d'application ; les étapes 2 à 4 sont effectuées par les gestionnaires de files d'attente locaux et les programmes MCA. L'application d'insertion ne connaît pas l'emplacement de la file d'attente cible, qui peut se trouver dans le même processeur ou dans un autre processeur sur un autre continent.

La combinaison de l'agent MCA émetteur, de la connexion réseau et de l'agent MCA récepteur est appelée *canal de transmission de messages* et est intrinsèquement une unité unidirectionnelle. Normalement, il est nécessaire de déplacer les messages dans les deux sens, et deux canaux sont établis pour ce mouvement, un dans chaque sens.

Tâches associées

Insertion de messages dans des files d'attente éloignées

Qu'est-ce que la résolution de nom de file d'attente?

La résolution de nom de file d'attente est vitale pour la gestion des files d'attente réparties. Il élimine la nécessité pour les applications de se préoccuper de l'emplacement physique des files d'attente et isole les applications des détails des réseaux.

Un administrateur système peut déplacer des files d'attente d'un gestionnaire de files d'attente à un autre et modifier le routage entre les gestionnaires de files d'attente sans que les applications aient besoin de connaître quoi que ce soit à ce sujet.

Pour dissocier de la conception de l'application le chemin exact sur lequel transitent les données, il existe un niveau d'indirection entre le nom utilisé par l'application lorsqu'elle fait référence à la file d'attente cible et le nom du canal sur lequel se produit le flux. Cette indirection est réalisée à l'aide du mécanisme de résolution de nom de file d'attente.

En substance, lorsqu'une application fait référence à un nom de file d'attente, ce nom est mappé par le mécanisme de résolution à une file d'attente de transmission ou à une file d'attente locale qui n'est pas une file d'attente de transmission. Pour le mappage à une file d'attente de transmission, une deuxième résolution de nom est nécessaire à la destination et le message reçu est placé dans la file d'attente cible comme prévu par le concepteur d'applications. L'application reste inconsciente de la file d'attente de transmission et du canal utilisés pour le déplacement du message.

Remarque : La définition de la file d'attente et du canal est une responsabilité de gestion du système et peut être modifiée par un opérateur ou un utilitaire de gestion du système, sans qu'il soit nécessaire de modifier les applications.

Une exigence importante pour la gestion système des flux de messages est que des chemins alternatifs doivent être fournis entre les gestionnaires de files d'attente. Par exemple, les exigences métier peuvent imposer que différentes *classes de service* soient envoyées via différents canaux à la même destination. Cette décision est une décision de gestion de système et le mécanisme de résolution de nom de file d'attente offre un moyen flexible de l'atteindre. Le guide de programmation d'application décrit cela en détail, mais l'idée de base est d'utiliser la résolution de nom de file d'attente au niveau du gestionnaire de files d'attente d'envoi pour mapper le nom de file d'attente fourni par l'application à la file d'attente de transmission appropriée pour le type de trafic impliqué. De même, à l'extrémité réceptrice, la résolution de nom de file d'attente mappe le nom du descripteur de message à une file d'attente locale (et non à une file d'attente de transmission) ou encore à une file d'attente de transmission appropriée.

Non seulement il est possible que le chemin de réacheminement d'un gestionnaire de files d'attente à un autre soit partitionné en différents types de trafic, mais le message de retour envoyé à la définition de file d'attente de réponse dans le message sortant peut également utiliser le même partitionnement de trafic. La résolution de nom de file d'attente répond à cette exigence et le concepteur d'applications n'a pas besoin d'être impliqué dans ces décisions de partitionnement de trafic.

Le point où le mappage est effectué au niveau des gestionnaires de files d'attente d'envoi et de réception est un aspect important du fonctionnement de la résolution de nom. Ce mappage permet de mapper le nom de file d'attente fourni par l'application d'insertion à une file d'attente locale ou à une file d'attente de transmission au niveau du gestionnaire de files d'attente d'envoi, puis de le remapper à une file d'attente locale ou à une file d'attente de transmission au niveau du gestionnaire de files d'attente de réception.

Les messages de réponse des applications réceptrices ou des agents MCA ont la même résolution de nom, ce qui permet le routage de retour sur des chemins spécifiques avec des définitions de file d'attente sur tous les gestionnaires de files d'attente sur la route.

Comment les attributs d'objet de destination sont-ils résolus pour les alias, les files d'attente éloignées et les files d'attente de cluster?

Lorsque la résolution de nom est effectuée pour le compte d'un appel d'API d'application, les attributs affectant l'utilisation de l'objet sont résolus à partir d'une combinaison de l'objet nommé à l'origine, du "chemin" (voir «Résolution de nom de file d'attente», à la page 79) et de l'objet cible résolu. Dans un cluster de gestionnaires de files d'attente, l'"objet nommé" en question est la définition d'objet en cluster (file d'attente ou rubrique). Il s'agit d'un sous-ensemble des attributs d'objet partagés entre les gestionnaires de files d'attente et visibles via. Par exemple, **DISPLAY QCLUSTER**.

Lorsqu'un attribut peut être défini sur l'objet nommé ouvert par l'application, il est prioritaire. Par exemple, tous les attributs DEF* *** (persistance par défaut, priorité et réponse d'insertion asynchrone) peuvent être configurés sur des définitions d'alias et de file d'attente éloignée. Elles prennent effet lorsque l'alias ou la file d'attente éloignée est ouvert par une application, plutôt que par une file d'attente de destination ou une file d'attente de transmission résolue.

Les attributs conçus pour restreindre ou limiter l'interaction de l'application avec un objet cible ne peuvent généralement pas être définis sur l'objet nommé (définition de file d'attente éloignée ou alias). Par exemple, **MAXMSGL** et **MAXDEPTH** ne peuvent pas être définis sur une définition de file d'attente éloignée ou un alias et ne sont pas transmis entre les membres d'un cluster de gestionnaires de files d'attente. Ces attributs sont donc extraits de la file d'attente résolue (par exemple, la file d'attente locale, la file d'attente de transmission appropriée ou **SYSTEM.CLUSTER.TRANSMIT.QUEUE**). Lors de l'arrivée dans un gestionnaire de files d'attente éloignées, une deuxième contrainte peut être appliquée lors de la livraison à la file d'attente cible, ce qui peut entraîner le placement d'un message dans une file d'attente de messages non livrés ou l'arrêt forcé du canal.

Notez qu'un cas particulier de résolution d'attribut est l'activation de **PUT** et **GET**. Pour ces deux attributs, toute instance de **DISABLED** dans le chemin de file d'attente génère un attribut résolu global de **DISABLED**.

Objets système et par défaut

Répertorie les objets système et par défaut créés par la commande **crtmqm**.

Lorsque vous créez un gestionnaire de files d'attente à l'aide de la commande de contrôle **crtmqm**, les objets système et les objets par défaut sont créés automatiquement.

- Les objets système sont les objets IBM MQ nécessaires au fonctionnement d'un gestionnaire de files d'attente ou d'un canal.
- Les objets par défaut définissent tous les attributs d'un objet. Lorsque vous créez un objet, tel qu'une file d'attente locale, tous les attributs que vous ne spécifiez pas explicitement sont hérités de l'objet par défaut.

Les tableaux suivants répertorient les objets système et par défaut créés par **crtmqm**.

Remarque : Deux autres objets par défaut ne sont pas inclus dans les tables: l'objet gestionnaire de files d'attente et le catalogue d'objets. Il s'agit d'objets dans le sens où ils sont consignés et récupérables.

- [Objets système et par défaut: files d'attente](#)
- [Objets système et par défaut: rubriques](#)
- [Objets système et par défaut: canaux serveur](#)
- [Objets système et par défaut: canaux client](#)
- [Objets système et par défaut: informations d'authentification](#)
- [Objets système et par défaut: informations de communication](#)
- [Objets système et par défaut: programmes d'écoute](#)
- [Objets système et par défaut: listes de noms](#)
- [Objets système et par défaut: processus](#)
- [Objets système et par défaut: services](#)

Nom d'objet	Description
SYSTEM.ADMIN.ACCOUNTING.QUEUE	File d'attente des données de message de comptabilité générées lorsqu'une application se déconnecte du gestionnaire de files d'attente.
SYSTEM.ADMIN.ACTIVITY.QUEUE	File d'attente contenant les messages de rapport d'activité renvoyés.
SYSTEM.ADMIN.CHANNEL.EVENT	File d'attente d'événements pour les canaux.
SYSTEM.ADMIN.COMMAND.EVENT	File d'attente d'événements pour les événements de commande.
SYSTEM.ADMIN.COMMAND.QUEUE	File d'attente de commandes d'administration. Utilisé pour les commandes MQSC distantes et les commandes PCF.
SYSTEM.ADMIN.CONFIG.EVENT	File d'attente d'événements pour les événements de configuration.
SYSTEM.ADMIN.LOGGER.EVENT	File d'attente d'événements pour les messages d'événement de consignateur (récepteur de journal).
SYSTEM.ADMIN.PERFM.EVENT	File d'attente d'événements pour les événements de performances.
SYSTEM.ADMIN.PUBSUB.EVENT	File d'attente d'événements liés à la publication / l'abonnement du système

Tableau 43. Objets système et par défaut: files d'attente (suite)

Nom d'objet	Description
SYSTEM.ADMIN.QMGR.EVENT	File d'attente d'événements pour les événements du gestionnaire de files d'attente.
SYSTEM.ADMIN.STATISTICS.QUEUE	File d'attente qui contient les données de surveillance des statistiques de MQI, de file d'attente et de canal.
SYSTEM.ADMIN.TRACE.ACTIVITY.QUEUE	File d'attente qui affiche l'activité de trace.
SYSTEM.ADMIN.TRACE.ROUTE.QUEUE	La file d'attente qui contient les messages de réponse de trace-route renvoyés.
SYSTEM.AMQP.COMMAND.QUEUE	File d'attente de commandes d'administration IBM MQ pour AMQP
SYSTEM.AUTH.DATA.QUEUE	File d'attente contenant les listes de contrôle d'accès pour le gestionnaire de files d'attente. Utilisé par le gestionnaire des droits d'accès aux objets (OAM).
SYSTEM.BROKER.ADMIN.STREAM	Flux d'administration pour l'interface de publication / abonnement en file d'attente
SYSTEM.BROKER.CONTROL.QUEUE	File d'attente de contrôle de l'interface de publication / abonnement.
SYSTEM.BROKER.DEFAULT.STREAM	Flux par défaut pour l'interface de publication / abonnement en file d'attente
SYSTEM.BROKER.INTER.BROKER.COMMUNICATIONS	File d'attente de communications de courtier à courtier.
SYSTEM.CHANNEL.INITQ	File d'attente d'initialisation de canal.
SYSTEM.CHANNEL.SYNCQ	File d'attente qui contient les données de synchronisation des canaux.
SYSTEM.CHLAUTH.DATA.QUEUE	File d'attente de données d'authentification de canal IBM MQ
SYSTEM.CICS.INITIATION.QUEUE	File d'attente d'initialisation CICS par défaut.
SYSTEM.CLUSTER.COMMAND.QUEUE	File d'attente utilisée pour transmettre des messages au gestionnaire de files d'attente de référentiel.
SYSTEM.CLUSTER.HISTORY.QUEUE	File d'attente utilisée pour stocker l'historique des informations d'état de cluster à des fins de service.
SYSTEM.CLUSTER.REPOSITORY.QUEUE	File d'attente utilisée pour stocker toutes les informations de référentiel.
SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE	File d'attente utilisée pour créer des files d'attente de transmission individuelles pour chaque canal émetteur de cluster.
SYSTEM.CLUSTER.TRANSMIT.QUEUE	File d'attente de transmission de tous les messages vers tous les clusters.
SYSTEM.DEAD.LETTER.QUEUE	File d'attente de rebut (message non distribué).
SYSTEM.DEFAULT.ALIAS.QUEUE	File d'attente alias par défaut.
SYSTEM.DEFAULT.INITIATION.QUEUE	File d'attente d'initialisation par défaut.
SYSTEM.DEFAULT.LOCAL.QUEUE	File d'attente locale par défaut.

Tableau 43. Objets système et par défaut: files d'attente (suite)

Nom d'objet	Description
SYSTEM.DEFAULT.MODEL.QUEUE	File d'attente modèle par défaut.
SYSTEM.DEFAULT.REMOTE.QUEUE	File d'attente éloignée par défaut.
SYSTEM.DOTNET.XARECOVERY.QUEUE	File d'attente de reprise IBM MQ .NET XA
SYSTEM.DURABLE.MODEL.QUEUE	File d'attente utilisée comme modèle pour les abonnements durables gérés.
SYSTEM.DURABLE.SUBSCRIBER.QUEUE	File d'attente utilisée pour stocker une copie persistante des abonnements durables dans le gestionnaire de files d'attente.
SYSTEM.HIERARCHY.STATE	Etat de la relation de hiérarchie de publication / abonnement répartie IBM MQ .
SYSTEM.INTER.QMGR.CONTROL	File d'attente de contrôle de publication / abonnement répartie IBM MQ .
SYSTEM.INTER.QMGR.FANREQ	File d'attente d'entrée du processus de distribution de l'abonnement au proxy interne de publication / abonnement distribué IBM MQ .
SYSTEM.INTER.QMGR.PUBS	Publications de publication / abonnement distribué IBM MQ .
SYSTEM.INTERNAL.REPLY.QUEUE	
SYSTEM.INTERNAL.REQUEST.QUEUE	
SYSTEM.JMS.TEMPQ.MODEL	Modèle pour les files d'attente temporaires JMS
SYSTEM.MQEXPLORER.REPLY.MODEL	File d'attente de réponse IBM MQ Explorer . Il s'agit d'une file d'attente modèle qui crée une file d'attente dynamique temporaire pour les réponses à IBM MQ Explorer.
SYSTEM.MQSC.REPLY.QUEUE	File d'attente de réponse de la commande MQSC. Il s'agit d'une file d'attente modèle qui crée une file d'attente dynamique temporaire pour les réponses aux commandes MQSC distantes.
SYSTEM.NDURABLE.MODEL.QUEUE	File d'attente utilisée comme modèle pour les abonnements non durables gérés.
SYSTEM.PENDING.DATA.QUEUE	Prise en charge des messages différés dans JMS.
SYSTEM.PROTECTION.ERROR.QUEUE	IBM MQ File d'attente d'erreurs de protection des messages.
SYSTEM.PROTECTION.POLICY.QUEUE	File d'attente des règles de protection des messages IBM MQ .
SYSTEM.REST.REPLY.QUEUE	
SYSTEM.RETAINED.PUB.QUEUE	File d'attente utilisée pour conserver une copie de chaque publication conservée dans le gestionnaire de files d'attente.
SYSTEM.SELECTION.EVALUATION.QUEUE	
SYSTEM.SELECTION.VALIDATION.QUEUE	

Tableau 44. Objets système et par défaut: rubriques

Nom d'objet	Description
SYSTEM.ADMIN.TOPIC	Rubrique d'administration.
SYSTEM.BASE.TOPIC	Rubrique de base pour la résolution ASPARENT . Si une rubrique particulière n'a pas d'objet de rubrique d'administration parent, ou si ces objets parent ont également ASPARENT, tous les attributs ASPARENT restants sont hérités de cet objet.
SYSTEM.BROKER.ADMIN.STREAM	Flux d'administration utilisé par l'interface de publication / abonnement en file d'attente.
SYSTEM.BROKER.DEFAULT.STREAM	Flux par défaut utilisé par l'interface de publication / abonnement en file d'attente.
SYSTEM.BROKER.DEFAULT.SUBPOINT	Sous-point par défaut utilisé par l'interface de publication / abonnement en file d'attente.
SYSTEM.DEFAULT.TOPIC	Définition de rubrique par défaut.

Tableau 45. Objets système et par défaut: canaux serveur

Nom d'objet	Description
SYSTEM.AUTO.RECEIVER	Canal récepteur dynamique.
SYSTEM.AUTO.SVRCONN	Canal de connexion serveur dynamique.
SYSTEM.DEF.AMQP	Canal AMQP par défaut. Notez que l'objet est défini, mais le service AMQP n'est pas pris en charge.
SYSTEM.DEF.CLUSRCVR	Canal récepteur par défaut du cluster, utilisé pour fournir des valeurs par défaut pour tous les attributs non spécifiés lorsqu'un canal CLUSRCVR est créé sur un gestionnaire de files d'attente du cluster.
SYSTEM.DEF.CLUSSDR	Canal émetteur par défaut pour le cluster, utilisé pour fournir des valeurs par défaut pour tous les attributs non spécifiés lorsqu'un canal CLUSSDR est créé sur un gestionnaire de files d'attente dans le cluster.
SYSTEM.DEF.RECEIVER	Canal récepteur par défaut.
SYSTEM.DEF.REQUESTER	Canal demandeur par défaut.
SYSTEM.DEF.SENDER	Canal émetteur par défaut.
SYSTEM.DEF.SERVER	Canal serveur par défaut.
SYSTEM.DEF.SVRCONN	Canal de connexion serveur par défaut.
SYSTEM.DEFAULT.AUTHINFO.IDPWLDAP	
SYSTEM.DEFAULT.AUTHINFO.IDPWOS	

Tableau 46. Objets système et par défaut: canaux client

Nom d'objet	Description
SYSTEM.DEF.CLNTCONN	Canal de connexion client par défaut.

Tableau 47. Objets système et par défaut: informations d'authentification

Nom d'objet	Description
SYSTEM.DEFAULT.AUTHINFO.CRLLDAP	Objet d'informations d'authentification par défaut permettant de définir des objets d'informations d'authentification de type CRLLDAP.
SYSTEM.DEFAULT.AUTHINFO.OCSP	Objet d'informations d'authentification par défaut permettant de définir des objets d'informations d'authentification de type OCSP.

Tableau 48. Objets système et par défaut: informations de communication

Nom d'objet	Description
SYSTEM.DEFAULT.COMMINFO.MULTICAST	Objet d'informations de communication par défaut pour la multidiffusion.

Tableau 49. Objets système et par défaut: programmes d'écoute




Nom d'objet	Description
SYSTEM.DEFAULT.LISTENER.TCP	Programme d'écoute par défaut pour le transport TCP.
 SYSTEM.DEFAULT.LISTENER.LU62	Programme d'écoute LU62 par défaut.
 SYSTEM.DEFAULT.LISTENER.NETBIOS	Programme d'écoute NETBIOS par défaut.
 SYSTEM.DEFAULT.LISTENER.SPX	Programme d'écoute SPX par défaut.

Tableau 50. Objets système et par défaut: listes de noms

Nom d'objet	Description
SYSTEM.DEFAULT.NAMELIST	Définition de liste de noms par défaut.
SYSTEM.QPUBSUB.QUEUE.NAMELIST	Liste des noms de file d'attente surveillés par l'interface de publication / abonnement en file d'attente.
SYSTEM.QPUBSUB.SUBPOINT.NAMELIST	Liste des objets de rubrique utilisés par l'interface de publication / abonnement en file d'attente pour faire correspondre des objets de rubrique à des points d'abonnement.

Tableau 51. Objets système et par défaut: processus

Nom d'objet	Description
SYSTEM.DEFAULT.PROCESS	Définition de processus par défaut.

Tableau 52. Objets système et par défaut: services

Nom d'objet	Description
SYSTEM.AMQP.SERVICE	Service d'API MQ Light . Notez que l'objet est défini, mais que le service n'est pas pris en charge.

Tableau 52. Objets système et par défaut: services (suite)

Nom d'objet	Description
SYSTEM.DEFAULT.SERVICE	Service par défaut.

SYSTEM.BASE.TOPIC

Rubrique de base pour la résolution ASPARENT. Si une rubrique particulière n'a pas d'objet de rubrique d'administration parent, ou si ces objets parent ont également ASPARENT, tous les attributs ASPARENT restants sont hérités de cet objet.

Tableau 53. Valeurs par défaut de SYSTEM.BASE.TOPIC

Paramètre	Valeur
TOPICSTR	"
V9.4.0 CAPEXPY	NOLIMIT
CLROUTE	DIRECT
Grappe	La valeur par défaut est une chaîne vide.
COMMINFO	SYSTEM.DEFAULT.COMMINFO.MULTICAST
DEFPRESP	SYNC
DEFPRTY	0
DEFPSIST	NO
DESCR	'Rubrique de base pour la résolution des attributs'
DURSUB	YES
MCAST	DISABLED
MDURMDL	SYSTEM.DURABLE.MODEL.QUEUE
MNDURMDL	SYSTEM.NDURABLE.MODEL.QUEUE
NPMSGDLV	ALLAVAIL
PMSGDLV	ALLDUR
PROXYSUB	FIRSTUSE
PUB	ENABLED
PUBSCOPE	ALL
z/OS z/OS QSGDISP (plateforme z/OS uniquement)	QMGR
SUB	ENABLED
SUBSCOPE	ALL
USEDLQ	YES
WILDCARD	PASSTHRU

Si cet objet n'existe pas, ses valeurs par défaut sont toujours utilisées par IBM MQ pour les attributs ASPARENT qui ne sont pas résolus par les rubriques parent plus haut dans l'arborescence de rubriques.

La définition des attributs PUB ou SUB de SYSTEM.BASE.TOPIC sur DISABLED empêche les applications de publier des rubriques ou de s'y abonner dans l'arborescence de rubriques, à deux exceptions près:

1. Tous les objets de rubrique de l'arborescence de rubriques dont PUB ou SUB est explicitement défini sur ENABLE. Les applications peuvent publier ou s'abonner à ces rubriques et à leurs enfants.
2. La publication et l'abonnement à SYSTEM.BROKER.ADMIN.STREAM ne sont pas désactivés par la définition des attributs PUB ou SUB de SYSTEM.BASE.TOPIC sur DISABLED.

Voir aussi [Traitement spécial](#) pour le paramètre **PUB**.

Informations sur les sections des fichiers de configuration

Les informations suivantes vous aident à configurer les informations dans des strophes et répertorient le contenu des fichiers `mqs.ini`, `qm.iniet` `mqcclient.ini`.




Configuration de sections

Utilisez les liens pour vous aider à configurer le ou les systèmes de votre entreprise:

- Les [strophes et attributs du fichier mqs.ini](#) permettent de configurer les éléments suivants:

- Section *AllQueueManagers*
- Section *DefaultQueueManager*
- Section *ExitProperties*
- Section *LogDefaults*
- Section *Sécurité* du fichier `qm.ini`

- Les [strophes et attributs du fichier qm.ini](#) vous aident à configurer:

-  Section *AccessMode* (Windows uniquement)
- Section *Service* -pour les services optionnels
- Section *Log*
-   Section *RestrictedMode* (systèmes AIX and Linux uniquement)
- Section *XAResourceManager*
- Sections *TCP*, *LU62* et *NETBIOS*
- Section *ExitPath*
- Section *QLErrorLog*
- Section *SSL*
- Section *ExitPropertiesLocal*

- La [configuration des services et des composants](#) vous aide à configurer les éléments suivants:

- Section *Service*
- Section *ServiceComponent*

et contient des liens vers la façon dont ils sont utilisés pour différents services sur les plateformes AIX, Linux, and Windows .



- La [configuration des exits API](#) permet de configurer les éléments suivants:

- Section *AllActivityTrace*
- Section *ApplicationTrace*

- La [configuration du comportement de trace d'activité](#) permet de configurer les éléments suivants:

- Section *ApiExitCommon*
- Section *ApiExitTemplate*
- Section *ApiExitLocal*

- Le fichier de configuration [IBM MQ MQI client](#), `mqcclient.ini`, vous aide à configurer les éléments suivants:

- Section *CHANNELS*
- Section *ClientExitPath*
-  Section *LU62*, *NETBIOS* et *SPX* (Windows uniquement)
- Section *MessageBuffer*
- Section *SSL*
- Section *TCP*
-  Section *Trace* (utilisée pour IBM MQ .NET et XMS .NET uniquement)
- «Sections du fichier de configuration pour la mise en file d'attente répartie», à la page 90 vous aide à configurer les éléments suivants:
 - Section *CHANNELS*
 - Section *TCP*
 - Section *LU62*
 - *NETBIOS*
 - Section *ExitPath*
- La définition des attributs de message de publication / abonnement en file d'attente permet de configurer les éléments suivants:
 - Attribut *PersistentPublishRetry*
 - Attribut *NonPersistentPublishRetry*
 - Attribut *Taille dePublishBatch*
 - Attribut *IntervallePublishRetry*

dans la section *Courtier* .





Avertissement : Vous devez créer une section *Courtier* si vous en avez besoin.

- L'utilisation de la configuration automatique permet de configurer les éléments suivants:
 - Section *AutoConfig*
 - Section *AutoCluster*
 - Section *Variables*

Fichiers de configuration

Voir :

- [Fichier *mqs.ini*](#)
- [Fichier *qm.ini*](#)
- [Fichier *mqclient.ini*](#)

pour obtenir la liste des sections possibles dans chaque fichier de configuration.  

Fichier *mqs.ini*

Exemple de fichier de configuration IBM MQ pour les systèmes AIX and Linux illustre un exemple de fichier *mqs.ini*.

Un fichier *mqs.ini* peut contenir les sections suivantes:

- [Gestionnaires *AllQueue*](#)
- [DefaultQueueGestionnaire](#)
- [ExitProperties](#)

- [LogDefaults](#)


En outre, il existe une section [QueueManager](#) pour chaque gestionnaire de files d'attente.

Fichier qm.ini

Exemple de fichier de configuration de gestionnaire de files d'attente pour les systèmes IBM MQ for AIX or Linux illustre un exemple de fichier qm . ini .

Un fichier qm . ini peut contenir les sections suivantes:

- [ExitPath](#)
- [Journal](#)
- [QMErrorLog](#)
- [QueueManager](#)
- [Sécurité](#)
- [ServiceComponent](#)

 Pour configurer [InstallableServices](#) , utilisez les sections [Service](#) et [ServiceComponent](#) .



- [Connexion](#) pour [DefaultBindType](#)



Avertissement : Vous devez créer une section [Connexion](#) si vous en avez besoin.

- [SSL et TLS](#)
- [TCP, LU62 et NETBIOS](#)
- [XAResourceManager](#)

En outre, vous pouvez modifier les éléments suivants:

-  [AccessMode](#) (Windows uniquement)
-  [RestrictedMode](#) (systèmes AIX and Linux uniquement)

à l'aide de la commande [crtmqm](#) .

Fichier mqclient.ini

Un fichier mqclient . ini peut contenir les sections suivantes:

- [CANAUX](#)
- [ClientExitChemin](#)
- [LU62, NETBIOS et SPX](#)
- [MessageBuffer](#)
- [SSL](#)
- [TCP](#)

En outre, il se peut que vous ayez besoin d'une [strophePreConnect](#) pour configurer un exit de préconnexion.

Sections du fichier de configuration pour la mise en file d'attente répartie

Description des sections du fichier de configuration du gestionnaire de files d'attente, qm.ini, relatives à la mise en file d'attente répartie.

Cette rubrique présente les sections du fichier de configuration du gestionnaire de files d'attente relatives à la mise en file d'attente répartie. Elle s'applique au fichier de configuration du gestionnaire de files d'attente pour IBM MQ for Multiplatforms. Le fichier est appelé qm . ini sur toutes les plateformes.

Les sections relatives à la mise en file d'attente répartie sont les suivantes:

- Canaux

- TCP
- LU62
- NETBIOS
- CHEMIN EXIT

Le [Figure 6](#), à la page 91 présente les valeurs que vous pouvez définir à l'aide de ces strophes. Lorsque vous définissez l'une de ces sections, vous n'avez pas besoin de démarrer chaque élément sur une nouvelle ligne. Vous pouvez utiliser un point-virgule (;) ou un caractère de hachage (#) pour indiquer un commentaire.

```
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=chlname   ; 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.
```

Figure 6. Sections *qm.ini* pour la mise en file d'attente répartie

Remarques :

1. EXITPATH s'applique uniquement aux plateformes suivantes:

-  AIX
-  Windows

Tâches associées

[Configuration](#)

 [Configuration de z/OS](#)

[Modification des informations de configuration IBM MQ sur Multiplatforms](#)

Attributs de canal

Cette section décrit les attributs de canal détenus dans les définitions de canal.

Vous choisissez les attributs d'un canal pour qu'ils soient optimaux pour un ensemble particulier de circonstances pour chaque canal. Toutefois, lorsque le canal est en cours d'exécution, les valeurs réelles peuvent avoir changé lors des négociations de démarrage. Voir [Préparation des canaux](#).

De nombreux attributs ont des valeurs par défaut et vous pouvez utiliser ces valeurs pour la plupart des canaux. Toutefois, dans les cas où les valeurs par défaut ne sont pas optimales, consultez cette section pour savoir comment sélectionner les valeurs correctes.

Remarque : Dans IBM MQ for IBM i, la plupart des attributs peuvent être spécifiés en tant que *SYSDFCTL, ce qui signifie que la valeur est extraite du canal par défaut du système dans votre système.

Les types de canal pour les attributs de canal IBM MQ sont répertoriés dans le tableau suivant, dans l'ordre des paramètres de la commande MQSC.

Remarque : Pour les canaux de cluster (les colonnes CLUSSDR et CLUSRCVR dans le tableau), si un attribut peut être défini sur les deux canaux, définissez-le sur les deux canaux en vous assurant que les paramètres sont identiques. S'il existe une différence entre les paramètres, ceux que vous spécifiez sur le canal CLUSRCVR seront probablement utilisés. Pour une explication, voir la rubrique sur les [canaux de cluster](#).

Tableau 54. Attributs de canal pour les types de canal


Zone d'attribut	Paramètre de commande MQSC	SDR	SVR	RcvR	RQST R	CLNT-CONN	RR-CONN	CLUS-DTS	CLUS-RCVR	AMQP
Affinité de connexion	AFFINITY					Oui				
Date d'alerte	ALTDAT	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
Heure de modification	ALTTIME	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
 AMQP actif	AMQPKA									Oui
Intervalle des pulsations par lots	BATCHHB	Oui	Oui					Oui	Oui	
Intervalle entre les lots	BATCHINT	Oui	Oui					Oui	Oui	
Limite du lot	BATCHLIM	Oui	Oui					Oui	Oui	
Taille des lots	BATCHSZ	Oui	Oui	Oui	Oui			Oui	Oui	
Label de certificat	CERTLABL	Oui	Oui	Oui	Oui	Oui	Oui	Oui ^{«1»} , à la page 96	Oui	Oui
Nom du canal	Canal	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
Type de canal	CHLTYPE	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
Poids du canal client	CLNTWGH T					Oui				

Tableau 54. Attributs de canal pour les types de canal (suite)

Zone d'attribut	Paramètre de commande MQSC	SDR	SVR	RCV R	RQST R	CLNT-CONN	RR-CONN	CLUS-DTS	CLUS-RCVR	AMQP
Liste de noms de cluster	CLUSNL							Oui	Oui	
Grappe	CLUSTER							Oui	Oui	
Priorité de charge de travail de cluster	CLWLPRTY							Oui	Oui	
Rang de charge de travail de cluster	CLWLRANK							Oui	Oui	
Poids de charge de travail de cluster	CLWLWGH T							Oui	Oui	
Compression d'en-tête	COMPHDR	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
Compression des données	COMPMSG	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
Nom de connexion	CONNNAME	Oui	Oui		Oui	Oui		Oui	Oui	
Message de conversion	CONVERT	Oui	Oui					Oui	Oui	
Reconnexion par défaut	DEFRECON					Oui				
Description	DESCR	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
Intervalle de déconnexion	DISCINT	Oui	Oui				<input checked="" type="checkbox"/> z/OS Oui <input checked="" type="checkbox"/> Multi Non	Oui	Oui	
Intervalle des pulsations	HBINT	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
Intervalle de signaux de présence	KAINT	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
Adresse locale	LOCLADDR	Oui	Oui		Oui	Oui		Oui	Oui	Oui
Nombre de relances longues	LONGRTY	Oui	Oui					Oui	Oui	
Intervalle entre relances longues	LONGTMR	Oui	Oui					Oui	Oui	
Nombre maximal d'instances	MAXINST						Oui			Oui

Tableau 54. Attributs de canal pour les types de canal (suite)

Zone d'attribut	Paramètre de commande MQSC	SDR	SVR	RcvR	RQST R	CLNT-CONN	RR-CONN	CLUS-DTS	CLUS-RCVR	AMQP
Nombre maximal d'instances par client	MAXINSTC						Oui			
Longueur maximale des messages	MAXMSGLE	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
Nom de l'agent MCA	MCANAME	Oui	Oui		Oui			Oui	Oui	
Type de l'agent MCA	MCTYPE	Oui	Oui		Oui			Oui	Oui	
Utilisateur de l'agent MCA	MCAUSER	Oui	Oui	Oui	Oui		Oui	Oui	Oui	Oui
Nom de mode LU 6.2	MODENAME	Oui	Oui		Oui	Oui		Oui	Oui	
Monitoring	MONCHL	Oui	Oui	Oui	Oui		Oui	Oui	Oui	
Données utilisateur de l'exit de relance de message	MRDATA			Oui	Oui				Oui	
Nom de l'exit de relance de message	MREXIT			Oui	Oui				Oui	
Nombre de relances de message	MRRTY			Oui	Oui				Oui	
Intervalle entre relances de message	MRTMR			Oui	Oui				Oui	
Données utilisateur d'exit de message	MSGDATA	Oui	Oui	Oui	Oui			Oui	Oui	
Nom d'exit de message	MSGEXIT	Oui	Oui	Oui	Oui			Oui	Oui	
Priorité de connexion réseau	NETPRTY								Oui	
Vitesse des messages non persistants	NPMSPEED	Oui	Oui	Oui	Oui			Oui	Oui	
Mot de passe	PASSWORD	Oui	Oui		Oui	Oui		Oui		
Numéro de port	PORT									Oui

Tableau 54. Attributs de canal pour les types de canal (suite)





Zone d'attribut	Paramètre de commande MQSC	SDR	SVR	RCV R	RQST R	CLNT-CONN	RR-CONN	CLUS-DTS	CLUS-RCVR	AMQP
<u>Options de canal</u> PROPCTL pour MQGMO	PROPCTL	Oui	Oui					Oui	Oui	
<u>Droit d'insertion</u>	PUTAUT			Oui	Oui		 z/OS Oui  Multi Non		Oui	
<u>Nom gest. de files</u>	QMNAME					Oui				
 z/OS <u>Disposition</u>	QSGDISP	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Données utilisateur d'exit de réception</u>	RCVDATA	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Nom d'utilisateur de l'exit de réception</u>	RCVEXIT	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Données utilisateur d'exit de sécurité</u>	SCYDATA	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Nom d'exit de sécurité</u>	SCYEXIT	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Données utilisateur d'exit d'émission</u>	SENDDAT A	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Nom d'exit d'émission</u>	SENDEXIT	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Valeur de bouclage du numéro de séquence</u>	SEQWRAP	Oui	Oui	Oui	Oui			Oui	Oui	
<u>Connexions partagées</u>	SHARECN V					Oui	Oui			
<u>Nombre de relances courtes</u>	SHORTRT Y	Oui	Oui					Oui	Oui	
<u>Intervalle entre relances courtes</u>	SHORTTM R	Oui	Oui					Oui	Oui	
 z/OS <u>Protection des règles de sécurité</u>	SPLPROT	Oui	Oui	Oui	Oui					
<u>Authentification client SSL</u>	SSLCAUTH		Oui	Oui	Oui		Oui		Oui	Oui

Tableau 54. Attributs de canal pour les types de canal (suite)

Zone d'attribut	Paramètre de commande MQSC	SDR	SVR	RCV R	RQST R	CLNT-CONN	RR-CONN	CLUS-DTS	CLUS-RCVR	AMQP
<u>Spécification de chiffrement SSL</u>	SSLCIPH	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
<u>Homologue SSL</u>	SSLPEER	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui
<u>Statistiques de canal</u>	STATCHL	Oui	Oui	Oui	Oui			Oui	Oui	
<u>Nom du programme de transaction LU 6.2</u>	TPNAME	Oui	Oui		Oui	Oui		Oui	Oui	
<u>Racine des rubriques</u>	TPROOT									Oui
<u>Type de transfert</u>	TRPTYPE	Oui	Oui	Oui	Oui	Oui	Oui	Oui	Oui	
<u>Utiliser l'ID client</u>	USECLTID									Oui
<u>Utiliser la file d'attente de rebut</u>	USEDLQ	Oui	Oui	Oui	Oui			Oui	Oui	
<u>ID utilisateur</u>	USERID	Oui	Oui		Oui	Oui		Oui		
<u>Nom de la file d'attente de transmission</u>	XMITQ	Oui	Oui							

Remarques :

1. Aucune des interfaces d'administration ne permet d'obtenir ou de définir cet attribut pour les canaux CLUSSDR. Vous recevrez un message MQRCCF_INJUSTEMENT _channel_type . Toutefois, l'attribut est présent dans les objets de canal CLUSSDR (y compris les structures MQCD) et un exit CHAD peut le définir à l'aide d'un programme si nécessaire.

IBM MQ pour certaines plateformes peut ne pas implémenter tous les attributs affichés dans cette section. Les exceptions et les différences de plateforme sont mentionnées dans les descriptions d'attribut individuelles, le cas échéant.

Le nom de chaque attribut est indiqué entre crochets.

Les attributs sont classés par ordre alphabétique en groupes.

Référence associée

Commandes MQSC

ALTER CHANNEL

De la définition d'un canal

Attributs de canal pour les mots clés MQSC (A-B)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par la lettre A ou B.

AFFINITY (affinité de connexion)

Cet attribut indique si les applications client qui se connectent plusieurs fois à l'aide du même nom de gestionnaire de files d'attente utilisent le même canal client.

Utilisez cet attribut (MQIACH_CONNECTION_AFFINITY) lorsque plusieurs définitions de canal applicables sont disponibles.

Les valeurs possibles sont les suivantes:

PREFERRED

La première connexion dans un processus de lecture d'une table de définition de canal du client (CCDT) crée une liste de définitions applicables basées sur la pondération du canal client, en classant les définitions de pondération nulle en premier et par ordre alphabétique. Chaque connexion du processus tente de se connecter en utilisant la première définition de la liste. Si la connexion échoue, la définition suivante est utilisée. Les définitions infructueuses dotées de valeurs de pondération de canal client autres que 0 sont placées en fin de liste. Les définitions pour lesquelles la pondération de canal client est égale à 0 restent en tête de liste et sont sélectionnées en premier pour chaque connexion.

Chaque processus client portant le même nom d'hôte crée toujours la même liste.

Pour les applications client écrites en C, C++ ou l'infrastructure de programmation .NET (y compris .NET entièrement gérée), et pour les applications qui utilisent IBM MQ classes for Java et IBM MQ classes for JMS, la liste est mise à jour si la table de définition de canal du client a été modifiée depuis la création de la liste.

Il s'agit de la valeur par défaut, qui a la valeur 1.

Aucun

La première connexion dans un processus de lecture d'une table de définition de canal du client (CCDT) crée une liste de définitions applicables. Toutes les connexions du processus sélectionnent une définition applicable basées sur la pondération du canal client, en sélectionnant les définitions de pondération nulle en premier et par ordre alphabétique.

Pour les applications client écrites en C, C++ ou l'infrastructure de programmation .NET (y compris .NET entièrement gérée), et pour les applications qui utilisent IBM MQ classes for Java et IBM MQ classes for JMS, la liste est mise à jour si la table de définition de canal du client a été modifiée depuis la création de la liste.

Cet attribut est valide uniquement pour le type de canal de connexion client.

ALTDATA (modification de la date)

Cet attribut est la date à laquelle la définition a été modifiée pour la dernière fois, au format yyyy-mm-dd. Il est valide pour tous les types de canal.

ALTTIME (Modification de l'heure)

Cet attribut correspond à l'heure à laquelle la définition a été modifiée pour la dernière fois, au format hh.mm.ss et est valide pour tous les types de canal.

AMQPKA (signal de présence AMQP)



Utilisez l'attribut **AMQPKA** pour spécifier un temps de signal de présence pour la connexion client AMQP. Si le client AMQP n'a pas envoyé de trames dans l'intervalle de signal de présence, la connexion est fermée.

L'attribut **AMQPKA** détermine la valeur de l'attribut idle-timeout envoyé de IBM MQ à un client AMQP. L'attribut correspond à une période en millisecondes.

Si **AMQPKA** est défini sur une valeur > 0, IBM MQ la moitié de cette valeur est transmise en tant qu'attribut idle-timeout. Par exemple, la valeur 10000 entraîne l'envoi par le gestionnaire de files d'attente d'une valeur de délai d'inactivité de 5000. Le client doit s'assurer que les données sont envoyées à IBM MQ au moins toutes les 10000 millisecondes. Si les données ne sont pas reçues par IBM MQ dans ce délai, IBM MQ suppose que le client a perdu sa connexion et force la fermeture de la connexion avec une condition d'erreur `amqp:resource-limit-exceeded`.

La valeur AUTO ou 0 signifie que IBM MQ ne envoie pas d'attribut de délai d'inactivité au client AMQP.

Un client AMQP peut toujours transmettre une valeur de délai d'inactivité qui lui est propre. Si tel est le cas, IBM MQ transite des données (ou un cadre AMQP vide) au moins aussi fréquemment pour informer le client qu'il est disponible.

BATCHHB (Intervalle des pulsations par lots)

son attribut permet à un canal émetteur de vérifier que le canal récepteur est toujours actif juste avant de valider un lot de messages.

L'intervalle des pulsations par lots permet ainsi d'éviter que le lot ne soit en attente de validation si le canal récepteur n'est pas actif. En procédant à l'annulation du lot, les messages restent disponibles pour être traités de sorte qu'ils puissent, par exemple, être redirigés vers un autre canal.

Si le canal émetteur a reçu une communication du canal récepteur au cours de l'intervalle des pulsations par lots, le canal récepteur est supposé être toujours actif, sinon une pulsation est envoyée au canal récepteur pour vérification. Le canal émetteur attend une réponse de l'extrémité réceptrice du canal pendant une période donnée, en fonction du nombre de secondes spécifié dans l'attribut de canal Intervalle des pulsations (HBINT).

La valeur est exprimée en millisecondes et doit être comprise entre zéro et 999999. La valeur zéro indique que le signal de présence du lot n'est pas utilisé.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

BATCHINT (Intervalle par lots)

Cet attribut est une période, en millisecondes, pendant laquelle le canal maintient un lot ouvert même s'il n'y a pas de messages dans la file d'attente de transmission.

Vous pouvez indiquer n'importe quel nombre de millisecondes, de zéro à 999 999 999. La valeur par défaut est zéro.

Si vous ne spécifiez pas d'intervalle de traitement par lots, le lot se ferme lorsque l'une des conditions suivantes est remplie:

- Le nombre de messages indiqué dans BATCHSZ a été envoyé.
- Le nombre d'octets indiqué dans BATCHLIM a été envoyé.
- La file d'attente de transmission est vide.

Sur les canaux à faible charge, où la file d'attente de transmission devient fréquemment vide, la taille de lot effective peut être beaucoup plus petite que BATCHSZ.

Vous pouvez utiliser l'attribut BATCHINT pour rendre vos canaux plus efficaces en réduisant le nombre de lots courts. Sachez toutefois que vous pouvez ralentir le temps de réponse, car les lots durent plus longtemps et les messages restent non validés plus longtemps.

Si vous spécifiez un BATCHINT, les lots ne se ferment que lorsque l'une des conditions suivantes est remplie:

- Le nombre de messages indiqué dans BATCHSZ a été envoyé.
- Le nombre d'octets indiqué dans BATCHLIM a été envoyé.
- Il n'y a plus de messages dans la file d'attente de transmission et un intervalle de temps de BATCHINT s'est écoulé lors de l'attente des messages (depuis que le premier message du lot a été extrait).

Remarque : BATCHINT indique le temps total passé à attendre des messages. Elle n'inclut pas le temps passé à extraire des messages qui sont déjà disponibles dans la file d'attente de transmission, ni le temps passé à transférer des messages.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

BATCHLIM (Limite du lot)

Cet attribut correspond à la limite, en kilooctets, de la quantité de données pouvant être envoyées via un canal avant de prendre un point de synchronisation.

Un point de synchronisation est défini après le passage dans le canal du message qui a entraîné l'atteinte de la limite.

La valeur doit être comprise entre 0 et 999999. La valeur par défaut est 5000.

La valeur zéro dans cet attribut signifie qu'aucune limite de données n'est appliquée aux lots sur ce canal.

Le lot est arrêté lorsque l'une des conditions suivantes est remplie:

- Des messages BATCHSZ ont été envoyés.
- Les octets BATCHLIM ont été envoyés.
- La file d'attente de transmission est vide et la valeur de BATCHINT est dépassée.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

BATCHSZ (Taille de lot)

Cet attribut correspond au nombre maximal de messages à envoyer avant la prise d'un point de synchronisation.

La taille du lot n'affecte pas la façon dont le canal transfère les messages ; les messages sont toujours transférés individuellement, mais sont validés ou annulés en tant que lot.

Pour améliorer les performances, vous pouvez définir une taille de lot pour définir le nombre maximal de messages à transférer entre deux *points de synchronisation*. La taille de lot à utiliser est négociée lorsqu'un canal démarre et la valeur la plus basse des deux définitions de canal est utilisée. Dans certaines implémentations, la taille de lot est calculée à partir de la plus basse des deux définitions de canal et des deux valeurs MAXUMSGS du gestionnaire de files d'attente. La taille réelle d'un lot peut être inférieure ; par exemple, un lot se termine lorsqu'il ne reste aucun message dans la file d'attente de transmission ou que l'intervalle de traitement par lots expire.

Une valeur élevée pour la taille de lot augmente le débit, mais les temps de récupération sont augmentés car il y a plus de messages à renvoyer et à envoyer à nouveau. La valeur par défaut de BATCHSZ est 50 et il est conseillé d'essayer d'abord cette valeur. Vous pouvez choisir une valeur inférieure pour BATCHSZ si vos communications ne sont pas fiables, ce qui rend la reprise plus probable.

La procédure de point de synchronisation a besoin d'un identificateur d'unité d'oeuvre logique unique à échanger sur la liaison à chaque fois qu'un point de synchronisation est pris, afin de coordonner les procédures de validation par lots.

Si la procédure de validation par lots synchronisée est interrompue, une situation *en attente de validation* peut se produire. Les situations en attente de validation sont résolues automatiquement lorsqu'un canal de transmission de messages démarre. Si cette résolution échoue, une intervention manuelle peut être nécessaire à l'aide de la commande RESOLVE.

Quelques remarques à prendre en compte lors du choix du nombre pour la taille de lot:

- Si ce nombre est trop élevé, la quantité d'espace de file d'attente utilisée aux deux extrémités de la liaison devient excessive. Les messages prennent de l'espace en file d'attente lorsqu'ils ne sont pas validés et ne peuvent pas être supprimés des files d'attente tant qu'ils ne sont pas validés.
- S'il est probable qu'il y ait un flux régulier de messages, vous pouvez améliorer les performances d'un canal en augmentant la taille du lot car moins de flux de confirmation sont nécessaires pour transférer la même quantité d'octets.
- Si les caractéristiques du flux de messages indiquent que les messages arrivent par intermittence, une taille de lot de 1 avec un intervalle de temps de déconnexion relativement important peut offrir de meilleures performances.
- Ce nombre peut être compris entre 1 et 9999.
- Même si les messages non persistants sur un canal rapide n'attendent pas de point de synchronisation, ils contribuent au nombre de tailles de lot.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

Attributs de canal pour les mots clés MQSC (C)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par la lettre C.

CERTLABL (label de certificat)

Cet attribut spécifie le libellé de certificat de la définition de canal.

Le libellé identifie le certificat personnel dans le référentiel de clés qui est envoyé à l'homologue distant. Le certificat est défini comme décrit dans [Labels de certificat numérique](#).

Les canaux entrants (y compris les canaux RCVR, RQSTR, CLUSRCVR, SERVER non qualifié et SVRCONN) n'envoient le certificat configuré que si la version IBM MQ de l'homologue distant prend entièrement en charge la configuration de libellé de certificat et que le canal utilise un CipherSpecTLS.

Si tel n'est pas le cas, l'attribut **CERTLABL** du gestionnaire de files d'attente détermine le certificat envoyé. Cette restriction est due au fait que le mécanisme de sélection de libellé de certificat pour les canaux entrants dépend d'une extension de protocole TLS qui n'est pas prise en charge dans tous les cas. En particulier, les clients Java et JMS ne prennent pas en charge l'extension de protocole requise et ne reçoivent que le certificat configuré par l'attribut **CERTLABL** du gestionnaire de files d'attente, quel que soit le paramètre de libellé spécifique au canal.

Un canal serveur non qualifié est un canal dont la zone CONNAME n'est pas définie.

Aucune des interfaces d'administration ne permet d'obtenir ou de définir cet attribut pour les canaux CLUSSDR. Vous recevrez un message MQRCCF_IL LIC_CHANNEL_TYPE. Toutefois, l'attribut est présent dans les objets de canal CLUSSDR (y compris les structures MQCD) et un exit CHAD peut le définir à l'aide d'un programme si nécessaire.

Pour plus d'informations sur ce que le libellé de certificat peut contenir, voir [Labels de certificat numérique, compréhension des exigences](#).

Cet attribut est valide pour tous les types de canal.

Remarque : Pour SSL/TLS, CERTLABL doit être défini sur la définition QMGR. Vous pouvez éventuellement définir un CERTLABL sur la définition CHANNEL.

Le gestionnaire de files d'attente CERTLABL est vérifié et doit être un certificat personnel valide, même si vous définissez un CERTLABL sur la définition CHANNEL.

CHANNEL (nom du canal)

Cet attribut indique le nom de la définition de canal.

Le nom peut contenir jusqu'à 20 caractères, bien que les deux extrémités d'un canal de transmission de messages doivent avoir le même nom et que d'autres implémentations puissent avoir des restrictions sur la taille, le nombre réel de caractères doit être inférieur.

Dans la mesure du possible, les noms de canal sont uniques pour un canal entre deux gestionnaires de files d'attente d'un réseau de gestionnaires de files d'attente interconnectés.

Le nom doit contenir des caractères de la liste suivante:

Ordre alphabétique	(A-Z, a-z; notez que les majuscules et les minuscules sont significatives)
Caractères numériques	(0 à 9)
Période	(.)
Barre oblique	(/)
Trait de soulignement	(_)
Signe pourcentage	(%)

Remarque :

1. Les blancs imbriqués ne sont pas admis et les blancs de début sont ignorés.
2. Sur les systèmes utilisant EBCDIC Katakana, vous ne pouvez pas utiliser de caractères minuscules.

Cet attribut est valide pour tous les types de canal.

CHLTYPE (Type de canal)

Cet attribut indique le type du canal en cours de définition.


Les types de canal possibles sont les suivants:

Types de canaux de transmission de messages:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Emetteur de cluster
- Récepteur de cluster

Types de canal MQI:

- Connexion client (AIX, Linux, and Windows uniquement)

Remarque :  Les canaux de connexion client peuvent également être définis sur z/OS pour être utilisés sur d'autres plateformes.

- Connexion serveur
- AMQP

Les deux extrémités d'un canal doivent avoir le même nom et avoir des types compatibles:

- Expéditeur avec récepteur
- Demandeur avec serveur
- Demandeur avec expéditeur (pour rappel)
- Serveur avec récepteur (le serveur est utilisé comme expéditeur)
- Connexion client avec connexion serveur
- Émetteur de cluster avec récepteur de cluster
- AMQP avec AMQP

CLNTWGHT (pondération du canal client)

Cet attribut spécifie une pondération pour l'influence de la définition de canal de connexion client utilisée.

L'attribut de pondération du canal client est utilisé pour que les définitions de canal client puissent être sélectionnées de manière aléatoire en fonction de leur pondération lorsque plusieurs définitions sont disponibles.

Lorsqu'un client émet une demande MQCONN de connexion à un groupe de gestionnaires de files d'attente, en spécifiant un nom de gestionnaire de files d'attente commençant par un astérisque, qui active l'équilibrage de la pondération du client sur plusieurs gestionnaires de files d'attente, et que plusieurs définitions de canal appropriées sont disponibles dans la table de définition de canal du client (CCDT), la définition à utiliser est sélectionnée de manière aléatoire en fonction de la pondération, avec toutes les définitions CLNTWGHT (0) applicables sélectionnées en premier par ordre alphabétique.

Remarque : Lorsqu'une table de définition de canal du client JSON est utilisée, il est possible d'avoir plusieurs canaux portant le même nom. S'il existe plusieurs canaux portant le même nom et qu'ils possèdent CLNTWGHT (0), les canaux sont sélectionnés dans l'ordre dans lequel ils sont définis dans la table de définition de canal du client JSON.

Spécifiez une valeur comprise entre 0 et 99. La valeur par défaut est 0.

Elle indique qu'aucun équilibrage de charge n'est effectué et que les définitions applicables sont sélectionnées par ordre alphabétique. Pour autoriser un équilibrage de charge, choisissez une valeur comprise entre 1 et 99, où 1 est la pondération la plus faible et 99 la plus élevée. La répartition des connexions entre deux ou plusieurs canaux avec des pondérations non nulles est proportionnelle au rapport de ces pondérations. Par exemple, trois canaux avec des valeurs CLNTWGHT de 2, 4 et 14 sont sélectionnés à environ 10%, 20% et 70% du temps. Cette distribution n'est pas garantie. Si l'attribut AFFINITY de la connexion est défini sur PREFERRED, la première connexion choisit une définition de canal en fonction des pondérations du client, puis les connexions suivantes continuent d'utiliser la même définition de canal.

Cet attribut est valide uniquement pour le type de canal de connexion client.

CLUSNL (liste de noms de cluster)

Cet attribut est le nom de la liste de noms qui spécifie une liste de clusters auxquels appartient le canal.

Jusqu'à l'une des valeurs résultantes de CLUSTER ou CLUSNL peut être non vide. Si l'une des valeurs n'est pas à blanc, l'autre doit être à blanc.

Cet attribut est valide pour les types de canal suivants:

- Émetteur de cluster
- Récepteur de cluster

Cluster (CLUSTER)

Cet attribut correspond au nom du cluster auquel appartient le canal.

La longueur maximale est de 48 caractères, conformément aux règles de dénomination des objets IBM MQ .

Jusqu'à l'une des valeurs résultantes de CLUSTER ou CLUSNL peut être non vide. Si l'une des valeurs n'est pas à blanc, l'autre doit être à blanc.

Cet attribut est valide uniquement pour les types de canal suivants:

- Émetteur de cluster
- Récepteur de cluster

CLWLPRTY (Priorité de charge de travail de cluster)

L'attribut de canal CLWLPRTY indique l'ordre de priorité des canaux pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 0 et 9, où 0 correspond à la priorité la plus basse et 9 à la priorité la plus élevée.

Utilisez l'attribut de canal CLWLPRTY pour définir un ordre de priorité pour les destinations de cluster disponibles. IBM MQ sélectionne les destinations ayant la priorité la plus élevée avant de sélectionner les destinations ayant la priorité de destination de cluster la plus faible. S'il existe plusieurs destinations avec la même priorité, elle sélectionne la destination la moins récemment utilisée.

S'il existe deux destinations possibles, vous pouvez utiliser cet attribut pour autoriser la reprise en ligne. Les messages sont envoyés au gestionnaire de files d'attente avec le canal dont la priorité est la plus élevée. S'il devient indisponible, les messages passent au gestionnaire de files d'attente de priorité supérieure suivant. Les gestionnaires de files d'attente de priorité inférieure agissent comme des réserves.

IBM MQ vérifie le statut des canaux avant de les hiérarchiser. Seuls les gestionnaires de files d'attente disponibles sont candidats à la sélection.

Remarques :

- Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).
- La disponibilité d'un gestionnaire de files d'attente éloignées dépend du statut du canal de ce gestionnaire de files d'attente. Lorsque les canaux démarrent, leur état change plusieurs fois, certains d'entre eux étant moins préférentiels à l'algorithme de gestion de la charge de travail du cluster. En pratique, cela signifie que les destinations de priorité inférieure (sauvegarde) peuvent être choisies alors que les canaux vers les destinations de priorité supérieure (principale) sont en cours de démarrage.
- Si vous devez vous assurer qu'aucun message n'est acheminé vers une destination de sauvegarde, n'utilisez pas CLWLPRTY. Envisagez d'utiliser des files d'attente distinctes ou CLWLRANK avec un basculement manuel du serveur principal vers le serveur de sauvegarde.

CLWLRANK (rang de charge de travail de cluster)

L'attribut de canal **CLWLRANK** indique le rang des canaux pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 0 et 9, où 0 correspond au rang le plus bas et 9 au rang le plus élevé.

Utilisez l'attribut de canal **CLWLRANK** si vous souhaitez contrôler la destination finale des messages envoyés à un gestionnaire de files d'attente dans un autre cluster. Contrôlez le choix de la destination finale en définissant le rang des canaux qui connectent un gestionnaire de files d'attente aux gestionnaires de files d'attente de passerelle à l'intersection des clusters.

Lorsque vous définissez **CLWLRANK**, les messages empruntent une route spécifique via les clusters interconnectés vers une destination de rang supérieur. Par exemple, les messages parviennent à un gestionnaire de files d'attente de passerelle qui peut les envoyer à l'un des deux gestionnaires de files d'attente à l'aide des canaux classés 1 et 2. Ils sont envoyés automatiquement au gestionnaire de

files d'attente connecté par un canal ayant le rang le plus élevé, dans ce cas le canal est envoyé au gestionnaire de files d'attente classé 2.

IBM MQ obtient le rang des canaux avant de vérifier le statut des canaux. L'obtention du rang avant la vérification de l'état du canal signifie que même les canaux non accessibles sont disponibles pour la sélection. Il permet aux messages d'être acheminés via le réseau même si la destination finale n'est pas disponible.

Remarques :

- Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).
- Si vous avez également utilisé l'attribut de priorité **CLWLPRTY**, IBM MQ effectue une sélection entre les destinations disponibles. Si un canal n'est pas disponible pour la destination dont le rang est le plus élevé, le message est conservé dans la file d'attente de transmission. Il est libéré lorsque le canal devient disponible. Le message n'est pas envoyé à la destination disponible suivante dans l'ordre de classement.

CLWLWGHT (pondération de charge de travail du cluster)

L'attribut de canal CLWLWGHT indique le poids appliqué aux canaux CLUSSDR et CLUSRCVR pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 1 et 99, où 1 correspond à la pondération la plus faible et 99 à la pondération la plus élevée.

Utilisez CLWLWGHT pour envoyer des serveurs avec plus de messages de puissance de traitement. Plus le poids du canal est élevé, plus le nombre de messages envoyés via ce canal est élevé.

Remarques :

- Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).
- Lorsque CLWLWGHT est modifié à partir de la valeur par défaut de 50 sur n'importe quel canal, l'équilibrage de la charge de travail dépend du nombre total de fois où chaque canal a été choisi pour un message envoyé à n'importe quelle file d'attente en cluster. Pour plus d'informations, voir [«Algorithme de gestion de la charge de travail du cluster»](#), à la page 152.

COMPHDR (compression d'en-tête)

Cet attribut est une liste de techniques de compression de données d'en-tête prises en charge par le canal.

Pour les canaux émetteur, serveur, émetteur de cluster, récepteur de cluster et de connexion client, les valeurs spécifiées sont classées par ordre de préférence avec la première technique de compression prise en charge par l'extrémité distante du canal utilisé. Les techniques de compression mutuellement prises en charge par les canaux sont transmises à l'exit de message du canal émetteur où la technique de compression utilisée peut être modifiée par message. La compression modifie les données transmises aux exits d'envoi et de réception.

Les valeurs possibles sont les suivantes :

Aucun

Aucune compression de données d'en-tête n'est effectuée. Cette valeur est la valeur par défaut.

SYSTEME

La compression de données d'en-tête est effectuée.

Cet attribut est valide pour tous les types de canal.

COMPMSG (compression de données)

Cet attribut est une liste des techniques de compression de données de message prises en charge par le canal.

Pour les canaux émetteur, serveur, émetteur de cluster, récepteur de cluster et de connexion client, les valeurs indiquées sont classées par ordre de préférence. La première technique de compression prise en charge par l'extrémité éloignée du canal est utilisée. Les techniques de compression mutuellement prises en charge par les canaux sont transmises à l'exit de message du canal émetteur où la technique de compression utilisée peut être modifiée par message. La compression modifie les données transmises aux exits d'envoi et de réception. Voir «[COMPHDR \(compression d'en-tête\)](#)», à la page 104 pour la compression de l'en-tête de message.

Les valeurs possibles sont les suivantes:

Aucun

Aucune compression de données de message n'est effectuée. Cette valeur est la valeur par défaut.

RLE

La compression de données de message est effectuée à l'aide de l'algorithme RLE.

ZLIBFAST

La compression de données de message est effectuée à l'aide de la technique de compression zlib. Il est préférable d'utiliser une durée de compression rapide.

ZLIBFAST peut éventuellement être déchargé dans la fonction de compression de données zEnterprise . Pour plus d'informations, voir [zEDC Express facility](#) .

ZLIBHIGH

La compression de données de message est effectuée à l'aide de la technique de compression zlib. Il est préférable d'utiliser une compression de haut niveau.

V 9.4.0 LZ4FAST

La compression des données de message est effectuée à l'aide de la technique de compression LZ4. Il est préférable d'utiliser une durée de compression rapide.

V 9.4.0 LZ4HIGH

La compression des données de message est effectuée à l'aide de la technique de compression LZ4 . Il est préférable d'utiliser une compression de haut niveau.

ANY

Permet au canal de prendre en charge toute technique de compression prise en charge par le gestionnaire de files d'attente. Pris en charge uniquement pour les canaux de réception, de demande et de connexion serveur.

Cet attribut est valide pour tous les types de canal.

AIX Depuis IBM MQ 9.3.0, les techniques ZLIBFAST et ZLIBHIGH peuvent utiliser la bibliothèque zlibNX à accélération matérielle sous IBM MQ for AIX si elle est installée. La bibliothèque zlibNX est une version améliorée de la bibliothèque de compression zlib qui prend en charge la compression et la décompression de données par accélération matérielle à l'aide de coprocesseurs appelés accélérateurs Nest (NX) sur les serveurs à processeur IBM POWER9 . La bibliothèque zlibNX est disponible dans IBM AIX 7.2 avec Technology Level 4 Expansion Pack et versions ultérieures. Les messages hautement compressibles dont la taille est supérieure à 2KB sont les plus susceptibles de bénéficier de l'utilisation de la bibliothèque zlibNX , en réduisant l'utilisation de l'unité centrale. Pour permettre à un agent MCA d'utiliser la bibliothèque zlibNX , définissez la variable d'environnement AMQ_USE_ZLIBNX.

CONNNAME (nom de connexion)

Cet attribut correspond à l'identificateur de la connexion de communication. Il indique les liaisons de communication particulières à utiliser par ce canal.

Il est facultatif pour les canaux serveur, sauf si le canal serveur est déclenché, auquel cas il doit spécifier un nom de connexion.

Indiquez **CONNAME** comme liste de noms de machines séparés par des virgules pour le **TRPTYPE** indiqué. Généralement, un seul nom de machine est nécessaire. Vous pouvez fournir plusieurs noms de machine pour configurer plusieurs connexions avec les mêmes propriétés. Les connexions sont généralement testées dans l'ordre dans lequel elles sont spécifiées dans la liste des connexions jusqu'à ce qu'une connexion soit établie. La commande est modifiée pour les clients si l'attribut **CLNTWGHT** est fourni. Si aucune connexion n'aboutit, le canal tente à nouveau d'établir une connexion, conformément aux attributs du canal. Avec les canaux client, une liste de connexions offre une alternative à l'utilisation de groupes de gestionnaires de files d'attente pour configurer plusieurs connexions. Avec des canaux de message, une liste de connexions est utilisée pour configurer les connexions aux adresses alternatives d'un gestionnaire de files d'attente multi-instance.

Multi Sur [Multiplateformes](#), le paramètre de nom de connexion TCP/IP d'un canal récepteur de cluster est facultatif. Si vous laissez le nom de connexion vide, IBM MQ génère un nom de connexion automatiquement en utilisant le port par défaut et l'adresse IP actuelle du système. Vous pouvez remplacer le numéro de port par défaut, et continuer à utiliser l'adresse IP actuelle du système. Pour chaque nom de connexion laissez le nom IP à blanc, puis indiquez le numéro de port entre parenthèses, par exemple :

(1415)

Le fichier **CONNAME** généré est toujours dans le format décimal à points (IPv4) ou hexadécimal (IPv6), plutôt que sous la forme d'un nom d'hôte DNS alphanumérique.

La longueur maximale du nom dépend de la plateforme:

- **Multi** 264 caractères.
- **z/OS** 48 caractères (voir la [remarque 1](#)).

Si le type de transport est TCP

CONNAME est le nom d'hôte ou l'adresse réseau de la machine distante (ou la machine locale pour les canaux récepteurs de cluster). Par exemple, (ABC.EXAMPLE.COM), (2001:DB8:0:0:0:0:0:0) ou (127.0.0.1). Il peut inclure le numéro de port, par exemple (MACHINE(123)).

z/OS Il peut inclure le nom IP d'un groupe DNS dynamique ou un port d'entrée Network Dispatcher.

Si vous utilisez une adresse IPv6 dans un réseau qui ne prend en charge que IPv4, le nom de connexion n'est pas résolu. Dans un réseau qui utilise à la fois IPv4 et IPv6, le nom de connexion interagit avec l'adresse locale pour déterminer quelle pile IP est utilisée. Pour plus d'informations, voir «[LOCLADDR \(adresse locale\)](#)», à la page 111.

Si le type de transport est LU 6.2

Multi Si les paramètres TPNAME et MODENAME sont indiqués, indiquez le nom qualifié complet de l'unité logique partenaire. Si TPNAME et MODENAME sont à blanc, indiquez le nom de l'objet d'informations complémentaires CPI-C pour votre plateforme spécifique.

z/OS Il existe deux formes pour spécifier la valeur:

- Nom de l'unité logique

Il s'agit des informations d'unité logique associées au gestionnaire de files d'attente, qui regroupent le nom d'unité logique, le nom du programme transactionnel et le nom de mode en option. Ce nom peut être spécifié sous l'une des trois formes suivantes:

Tableau 55. Noms d'unité logique et formats	
Format	Exemple
nom ul	IGY12355
nom ul/nomTP	IGY12345/APING
nom ul/nomTP/nommode	IGY12345/APINGD/#INTER

Pour le premier format, le nom TP et le nom de mode doivent être indiqués pour les attributs TPNAME et MODENAME ; dans le cas contraire, ces attributs doivent être vides. Pour les canaux de connexion client, seul le premier format est autorisé.

- Nom symbolique

Il s'agit du nom de destination symbolique des informations d'unité logique associées au gestionnaire de files d'attente, tel qu'il est défini dans le fichier d'informations complémentaires. Les attributs TPNAME et MODENAME doivent être à blanc. Notez que, pour les canaux récepteurs de cluster, les informations complémentaires se trouvent sur les autres gestionnaires de files d'attente du cluster. Dans ce cas, il peut s'agir d'un nom qu'un exit de définition automatique de canal peut résoudre en informations d'unité logique appropriées pour le gestionnaire de files d'attente local.

Le nom d'unité logique indiqué ou implicite peut être celui d'un groupe de ressources générique VTAM.

Si le protocole de transmission est NetBIOS

CONNNAME est le nom NetBIOS défini sur la machine distante.

Si le protocole de transmission est SPX

CONNNAME est une adresse de style SPX composée d'une adresse réseau de 4 octets, d'une adresse de noeud de 6 octets et d'un numéro de socket de 2 octets. Entrez ces valeurs en hexadécimal, les adresses réseau et de noeud étant séparées par un point et le numéro de socket entre crochets.

Exemple :

```
CONNNAME('0a0b0c0d.804abcde23a1(5e86)')
```

Si le numéro de socket est omis, le numéro de socket IBM MQ SPX par défaut est utilisé. La valeur par défaut est X'5E86'.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur

Il est facultatif pour les canaux serveur, sauf si le canal serveur est déclenché, auquel cas il doit spécifier un nom de connexion.

- Demandeur
- Connexion client
- Émetteur de cluster
- Récepteur de cluster

Remarque :

1. Pour les longueurs de nom, vous pouvez contourner la limite de 48 caractères de l'une des manières suivantes:
 - Configurez vos serveurs DNS de manière à utiliser, par exemple, le nom d'hôte "monserveur" au lieu de "myserver.location.company.com", en vous assurant que vous pouvez utiliser le nom d'hôte abrégé.
 - Utilisez des adresses IP.
2. La définition du protocole de transmission est contenue dans [TRPTYPE](#).

CONVERT (Convertir le message)

Cet attribut indique que le message doit être converti dans le format requis par le système récepteur avant la transmission.

Les données de message d'application sont généralement converties par l'application de réception. Toutefois, si le gestionnaire de files d'attente éloignées se trouve sur une plateforme qui ne prend pas en charge la conversion de données, utilisez cet attribut de canal pour indiquer que le message doit être converti dans le format requis par le système récepteur **avant la transmission de** .

Les valeurs possibles sont yes et no. Si vous indiquez yes, les données d'application du message sont converties avant l'envoi si vous avez spécifié l'un des noms de format intégré ou si un exit de conversion de données est disponible pour un format défini par l'utilisateur (voir [Ecriture des exits de conversion de données](#)). Si vous spécifiez no, les données d'application du message ne sont pas converties avant l'envoi.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

Attributs de canal pour les mots clés MQSC (D-L)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par les lettres *D* à *L*.

DEFRECON (reconnexion par défaut)

Indique si une connexion client reconnecte automatiquement une application client si sa connexion est interrompue.

Les valeurs possibles sont les suivantes:

NO (par défaut)

A moins d'être remplacé par **MQCONNX**, le client n'est pas reconnecté automatiquement.

Oui

A moins d'être remplacé par **MQCONNX**, le client se reconnecte automatiquement.

QMGR

A moins d'être remplacé par **MQCONNX**, le client se reconnecte automatiquement, mais uniquement au même gestionnaire de files d'attente. L'option QMGR a le même effet que MQCNO_RECONNECT_Q_MGR.

Désactivé

La reconnexion est désactivée, même si elle est demandée par le programme client à l'aide de l'appel **MQCONNX** MQI.

Cet attribut est valide uniquement pour les canaux de connexion client.

DESCR (Description)

Cet attribut décrit la définition de canal et contient jusqu'à 64 octets de texte.

Remarque : Le nombre maximal de caractères autorisé est inférieur si le système utilise un jeu de caractères codés sur deux octets (DBCS).

Utilisez les caractères du jeu de caractères identifié par l'ID de jeu de caractères codés (CCSID) du gestionnaire de files d'attente pour vous assurer que le texte est correctement traduit s'il est envoyé à un autre gestionnaire de files d'attente.

Cet attribut est valide pour tous les types de canal.

DISCINT (intervalle de déconnexion)

Cet attribut correspond à la durée après laquelle un canal se ferme, si aucun message n'arrive pendant cette période.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Connexion serveur utilisant le protocole TCP uniquement
- Émetteur de cluster
- Récepteur de cluster

Cet attribut est un délai d'attente, exprimé en secondes.

Pour les canaux de message serveur à serveur (types de canal serveur, émetteur de cluster, émetteur et récepteur de cluster), l'intervalle est mesuré à partir du point de fin d'un lot, c'est-à-dire lorsque la taille du lot est atteinte ou lorsque l'intervalle de lot expire et que la file d'attente de transmission devient vide. Si aucun message n'arrive dans la file d'attente de transmission pendant l'intervalle de temps spécifié, le canal se ferme. (Le temps est approximatif.)

L'échange de données de contrôle de fermeture entre les deux extrémités du canal de message serveur à serveur comporte une indication de la raison de la fermeture. Cela permet de s'assurer que l'extrémité correspondante du canal reste disponible pour un redémarrage.

Vous pouvez indiquer un nombre de secondes compris entre zéro et 999 999, où la valeur zéro signifie qu'il n'y a pas de déconnexion ; attendez indéfiniment.

La valeur par défaut est de 6000 secondes (100 minutes) pour les canaux de transmission de messages serveur à serveur et de 0 (aucun délai d'attente) pour les canaux de connexion serveur. Vous pouvez modifier la valeur par défaut utilisée pour les nouveaux canaux que vous créez en modifiant les objets de canal par défaut. Par exemple, modifiez l'attribut DISCINT sur SYSTEM.DEF.SENDER pour définir une nouvelle valeur par défaut pour les nouveaux canaux d'envoi que vous définissez.

Pour les canaux de connexion serveur utilisant le protocole TCP, l'intervalle représente la valeur de déconnexion d'inactivité du client, en secondes. Si un programme de canal de connexion serveur n'a reçu aucune communication de la part de son client partenaire pendant cette durée, il met fin à la connexion.

L'intervalle d'inactivité de la connexion serveur s'applique entre les appels d'API IBM MQ provenant d'un client.

Remarque : Un MQGET à exécution potentiellement longue avec appel d'attente n'est pas classé comme inactif et, par conséquent, n'arrive jamais à expiration suite à l'expiration de DISCINT.



Avertissement : Les performances sont affectées par la valeur indiquée pour l'intervalle de déconnexion.

Une valeur faible (par exemple, quelques secondes) peut nuire aux performances du système en arrêtant et en redémarrant constamment le canal. Une valeur élevée (supérieure à une heure) peut signifier que les ressources système sont consommées sans avantage. Vous pouvez également spécifier un intervalle de pulsation, de sorte que lorsqu'il n'y a pas de message dans la file d'attente de transmission, l'agent MCA émetteur envoie un flux de pulsation à l'agent MCA récepteur, ce qui permet à l'agent MCA récepteur de mettre au repos le canal sans attendre l'expiration de l'intervalle de déconnexion. Pour que ces deux valeurs fonctionnent efficacement, la valeur de l'intervalle des pulsations doit être nettement inférieure à la valeur de l'intervalle des déconnexions.

La valeur DISCINT par défaut pour les canaux de message de serveur à serveur est de 6000 secondes (100 minutes). Cependant, une valeur de quelques minutes est souvent une valeur raisonnable à utiliser sans impact sur les performances ou pour que les canaux s'exécutent pendant des périodes de temps inutilement longues. Si cela est approprié pour votre environnement, vous pouvez modifier cette valeur, soit sur chaque canal individuel,

soit en modifiant l'attribut DISCONT dans les définitions de canal par défaut (par exemple, SYSTEM.DEF.SENDER pour les canaux émetteurs) avant de créer vos propres canaux.

Pour plus d'informations, voir [Arrêt et mise au repos des canaux](#).

HBINT (intervalle de pulsation)

Cet attribut indique la durée approximative entre les flux de pulsations qui doivent être transmis à partir d'un agent MCA (Message Channel Agent) d'envoi lorsqu'il n'y a pas de messages dans la file d'attente de transmission.

Les flux de pulsations débloquent l'agent MCA récepteur qui attend l'arrivée de messages ou l'expiration de l'intervalle de déconnexion. Lorsque l'agent récepteur MCA est débloqué, il peut déconnecter le canal sans attendre l'expiration du délai de déconnexion. Les flux de pulsations libèrent également les mémoires tampon qui ont été allouées pour les messages de grande taille et ferment les files d'attente qui ont été laissées ouvertes à l'extrémité réceptrice du canal.

La valeur est exprimée en secondes et doit être comprise entre 0 et 999 999. La valeur zéro signifie qu'aucun flux de pulsation ne doit être envoyé. La valeur par défaut est 300. Pour être plus utile, la valeur doit être nettement inférieure à la valeur de l'intervalle de déconnexion.

Avec les applications qui utilisent des API IBM MQ classes for Java, JMS ou .NET, la valeur HBINT est déterminée de l'une des manières suivantes:

- Soit par la valeur sur le canal SVRCONN utilisé par l'application.
- Ou par la valeur sur le canal CLNTCONN, si l'application a été configurée pour utiliser une table de définition de canal du client.

Pour les canaux de connexion serveur et de connexion client, les signaux de présence peuvent être transmis du côté serveur et du côté client indépendamment. Si aucune donnée n'a été transférée sur le canal pour l'intervalle des pulsations, l'agent MQI de connexion client envoie un flux de pulsations et l'agent MQI de connexion serveur y répond avec un autre flux de pulsations. Cela se produit quel que soit l'état du canal, par exemple, qu'il soit inactif lors d'un appel API ou qu'il soit inactif en attente d'une entrée utilisateur client. L'agent MQI de connexion serveur est également capable d'initier un signal de présence sur le client, quel que soit l'état du canal. Pour éviter que les agents MQI de connexion serveur et de connexion client ne se soient mutuellement transmis simultanément, le signal de présence du serveur est transmis après qu'aucune donnée n'a été transférée sur le canal pour l'intervalle des signaux de présence plus 5 secondes.

Pour les canaux de connexion serveur et de connexion client fonctionnant en mode canal avant IBM WebSphere MQ 7.0, les signaux de présence ne circulent que lorsqu'un agent MCA serveur attend une commande MQGET avec l'option WAIT spécifiée, qu'il a émise pour le compte d'une application client.


Pour plus d'informations sur le fonctionnement des canaux MQI dans les deux modes, voir [SharingConversations \(MQLONG\)](#).

KAINTE (intervalle de signal de présence)

Cet attribut est utilisé pour spécifier une valeur de délai d'attente pour un canal.

L'attribut Intervalle de signal de présence est une valeur transmise à la pile de communications en spécifiant le temps de signal de présence pour le canal. Il vous permet de spécifier une valeur de signal de présence différente pour chaque canal.

Vous pouvez définir l'attribut Intervalle de signal de présence (KAINTE) pour les canaux par canal.

 Sous Multiplateformes, vous pouvez accéder au paramètre et le modifier, mais il est uniquement stocké et transféré ; il n'y a pas d'implémentation fonctionnelle du paramètre. Si vous avez besoin de la fonctionnalité fournie par le paramètre KAINTE, utilisez le paramètre d'intervalle de pulsation (HBINT), comme décrit dans [«HBINT \(intervalle de pulsation\)»](#), à la page 110.

Pour que cet attribut ait un effet, le signal de présence TCP/IP doit être activé.

- **z/OS** Sous z/OS, vous activez le signal de présence en exécutant la commande MQSC ALTER QMGR TCPKEEP (YES).
- **Multi** Sous Multiplateformes, il se produit lorsque le paramètre KEEPALIVE=YES est spécifié dans la strophe TCP du fichier de configuration de la mise en file d'attente répartie, qm.ini, ou via IBM MQ Explorer.

Le signal de présence doit également être activé dans TCP/IP lui-même, à l'aide du fichier de configuration du profil TCP.

La valeur indique une heure, en secondes, et doit être comprise entre 0 et 99999. La valeur 0 de l'intervalle de signal de présence indique que le signal de présence spécifique au canal n'est pas activé pour le canal et que seule la valeur de signal de présence définie à l'échelle du système dans TCP/IP est utilisée. Vous pouvez également définir KAINTE sur la valeur AUTO (cette valeur est la valeur par défaut). Si KAINTE est défini sur AUTO, la valeur de Keepalive est basée sur la valeur de l'intervalle des pulsations négocié (HBINT) comme suit:

Tableau 56. Valeur HBINT négociée et valeur KAINTE correspondante	
HBINT négocié	KAINTE
>0	HBINT négocié + 60 secondes
0	0

Cet attribut est valide pour tous les types de canal.

La valeur est ignorée pour tous les canaux ayant un TransportType (TRPTYPE) autre que TCP ou SPX

LOCLADDR (adresse locale)

Cet attribut indique l'adresse de communication locale du canal.

Remarque : Les canaux AMQP ne prennent pas en charge le même format de LOCLADDR que les autres canaux IBM MQ . Pour plus d'informations, voir «[#unique_51/unique_51_Connect_42_locladdr_amqp](#)», à la page 114.

LOCLADDR pour tous les canaux à l'exception des canaux AMQP

Cet attribut s'applique uniquement si le type de transport (TRPTYPE) est TCP/IP. Pour tous les autres types de transport, il est ignoré.

Lorsqu'une valeur LOCLADDR est spécifiée, un canal qui est arrêté puis redémarré continue d'utiliser l'adresse TCP/IP spécifiée dans LOCLADDR. Dans les scénarios de reprise, cet attribut peut être utile lorsque le canal communique via un pare-feu. Elle est utile car elle supprime les problèmes causés par le redémarrage du canal avec l'adresse IP de la pile TCP/IP à laquelle elle est connectée. LOCLADDR peut également forcer un canal à utiliser une pile IPv4 ou IPv6 sur un système à double pile, ou une pile en mode double sur un système à pile unique.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Demandeur
- Connexion client
- Émetteur de cluster
- Récepteur de cluster

Lorsque LOCLADDR inclut une adresse réseau, celle-ci doit être une adresse réseau appartenant à une interface réseau sur le système sur lequel le canal est exécuté. Par exemple, lors de la définition d'un

canal émetteur sur le gestionnaire de files d'attente ALPHA au gestionnaire de files d'attente BETA à l'aide de la commande MSQC suivante:

```
DEFINE CHANNEL(TO.BETA) CHLTYPE(SDR) CONNAME(192.0.2.0) XMITQ(BETA) LOCLADDR(192.0.2.1)
```

L'adresse LOCLADDR est l'IPv4 adresse 192.0.2.1. Ce canal émetteur s'exécute sur le système du gestionnaire de files d'attente ALPHA, de sorte que l'adresse IPv4 doit appartenir à l'une des interfaces réseau de son système.

La valeur est l'adresse IP facultative et le port facultatif ou la plage de ports utilisée pour les communications TCP/IP sortantes. Le format de ces informations est le suivant :

```
LOCLADDR([ip-addr] [(low-port[,high-port])][, [ip-addr] [(low-port[,high-port])]])
```

La longueur maximale de **LOCLADDR**, y compris les adresses multiples, est MQ_LOCAL_ADDRESS_LENGTH.

Si vous omettez **LOCLADDR**, une adresse locale est automatiquement allouée.

Notez que vous pouvez définir **LOCLADDR** pour un client C à l'aide de la table de définition de canal client (CCDT).

Tous les paramètres sont facultatifs. L'omission de la partie `adr-ip` de l'adresse permet d'activer la configuration d'un numéro de port fixe pour un pare-feu IP. L'omission du numéro de port permet de sélectionner un adaptateur de réseau particulier sans qu'il soit nécessaire d'identifier un numéro de port local unique. La pile TCP/IP génère un numéro de port unique.

Indiquez `[, [ip-addr] [(low-port[,high-port])]]` plusieurs fois pour chaque adresse locale supplémentaire. Utilisez plusieurs adresses locales si vous souhaitez spécifier un sous-ensemble d'adaptateurs de réseau local. Vous pouvez également utiliser `[, [ip-addr] [(low-port[,high-port])]]` pour représenter une adresse réseau locale particulière sur différents serveurs qui font partie d'une configuration de gestionnaire de files d'attente multi-instance.

adr-ip

`adr-ip` est indiqué dans l'un des trois formats suivants :

IPv4 en notation décimale

Exemple: 192.0.2.1

IPv6 en notation hexadécimale

Exemple: 2001:DB8:0:0:0:0:0:0

Format de nom d'hôte alphanumérique

Par exemple WWW.EXAMPLE.COM

port-inférieur et port-supérieur

`port-inférieur` et `port-supérieur` sont des numéros de port placés entre parenthèses.

Le tableau suivant montre comment le paramètre **LOCLADDR** peut être utilisé :

<i>Tableau 57. Exemples de la façon dont le paramètre LOCLADDR peut être utilisé</i>	
LOCLADDR	Explication
9.20.4.98	Le canal se lie à cette adresse localement
9.20.4.98, 9.20.4.99	Le canal se lie à l'une des adresses IP. L'adresse peut être deux adaptateurs de réseau sur un serveur ou à un adaptateur de réseau différent sur deux serveurs différents dans une configuration multi-instance.
9.20.4.98(1000)	Le canal se lie à cette adresse et au port 1000 localement
9.20.4.98(1000,2000)	Le canal se lie à cette adresse et utilise un port dans la plage 1000 - 2000 localement
(1000)	Le canal se lie au port 1000 localement

Tableau 57. Exemples de la façon dont le paramètre **LOCLADDR** peut être utilisé (suite)

LOCLADDR	Explication
(1000,2000)	Le canal se lie au port dans la plage 1000 - 2000 localement

Lorsqu'un canal est démarré, les valeurs spécifiées pour le nom de connexion (CONNAME) et l'adresse locale (LOCLADDR) déterminent la pile IP utilisée pour la communication. La pile IP utilisée est déterminée comme suit:

- Si le système ne dispose que d'une pile IPv4 configurée, la pile IPv4 est toujours utilisée. Si une adresse locale (LOCLADDR) ou un nom de connexion (CONNAME) est spécifié en tant qu'adresse réseau IPv6, une erreur est générée et le canal ne démarre pas.
- Si le système ne dispose que d'une pile IPv6 configurée, la pile IPv6 est toujours utilisée. Si une adresse locale (LOCLADDR) est spécifiée en tant qu'adresse réseau IPv4, une erreur est générée et le canal ne démarre pas. Sur les plateformes prenant en charge l'adressage mappé IPv6, si un nom de connexion (CONNAME) est spécifié en tant qu'adresse réseau IPv4, l'adresse est mappée à une adresse IPv6. Par exemple, xxx.xxx.xxx.xxx est mappé à ::ffff:xxx.xxx.xxx.xxx. L'utilisation d'adresses mappées peut nécessiter des traducteurs de protocole. Dans la mesure du possible, évitez d'utiliser des adresses mappées.
- Si une adresse locale (LOCLADDR) est spécifiée en tant qu'adresse IP pour un canal, la pile de cette adresse IP est utilisée. Si l'adresse locale (LOCLADDR) est spécifiée en tant que nom d'hôte correspondant aux adresses IPv4 et IPv6, le nom de connexion (CONNAME) détermine laquelle des piles est utilisée. Si l'adresse locale (LOCLADDR) et le nom de connexion (CONNAME) sont indiqués en tant que noms d'hôte résolus en adresses IPv4 et IPv6, la pile utilisée est déterminée par l'attribut de gestionnaire de files d'attente IPADDRV.
- Si le système comporte deux piles IPv4 et IPv6 configurées et qu'aucune adresse locale (LOCLADDR) n'est spécifiée pour un canal, le nom de connexion (CONNAME) spécifié pour le canal détermine la pile IP à utiliser. Si le nom de connexion (CONNAME) est spécifié en tant que nom d'hôte correspondant aux adresses IPv4 et IPv6, la pile utilisée est déterminée par l'attribut de gestionnaire de files d'attente IPADDRV.

Multi Sous **Multiplateformes**, vous pouvez définir une valeur d'adresse locale par défaut qui est utilisée pour tous les canaux émetteurs pour lesquels aucune adresse locale n'est définie. La valeur par défaut est définie en définissant la variable d'environnement MQ_LCLADDR avant de démarrer le gestionnaire de files d'attente. Le format de la valeur correspond à celui de l'attribut MQSC LOCLADDR.

Adresses locales avec canaux émetteurs de cluster

Les canaux émetteurs de cluster héritent toujours de la configuration du canal récepteur de cluster correspondant, comme défini sur le gestionnaire de files d'attente cible. Cela est vrai même s'il existe un canal émetteur de cluster défini localement du même nom, auquel cas la définition manuelle n'est utilisée que pour la communication initiale.

Pour cette raison, il n'est pas possible de dépendre du paramètre LOCLADDR défini dans le canal récepteur du cluster car il est probable que l'adresse IP ne soit pas détenue par le système sur lequel les émetteurs de cluster sont créés. Pour cette raison, le paramètre LOCLADDR sur le récepteur de cluster ne doit pas être utilisé sauf s'il existe une raison de restreindre uniquement les ports mais pas l'adresse IP pour tous les émetteurs de cluster potentiels, et il est connu que ces ports sont disponibles sur tous les systèmes où un canal émetteur de cluster peut être créé.

Si un cluster doit utiliser LOCLADDR pour obtenir les canaux de communication sortants à lier à une adresse IP spécifique, utilisez un exit de définition automatique de canal utilisez le LOCLADDR par défaut pour le gestionnaire de files d'attente lorsque cela est possible. Lors de l'utilisation d'un exit de canal, il force la valeur LOCLADDR de l'exit dans l'un des canaux CLUSSDR définis automatiquement.

Si vous utilisez un LOCLADDR autre que celui par défaut pour les canaux émetteurs de cluster via l'utilisation d'un exit ou d'une valeur par défaut, tout canal émetteur de cluster défini manuellement

correspondant, par exemple à un gestionnaire de files d'attente de référentiel complet, doit également avoir la valeur LOCLADDR définie pour activer la communication initiale sur le canal.

Remarque : Si le système d'exploitation renvoie une erreur de liaison pour le port fourni dans LOCLADDR (ou tous les ports, si une plage de ports est fournie), le canal ne démarre pas ; le système émet un message d'erreur.

LOCLADDR pour les canaux AMQP

Les canaux AMQP prennent en charge un format LOCLADDR différent de celui des autres canaux IBM MQ :

LOCLADDR (*ip-addr*)

LOCLADDR est l'adresse de communication locale du canal. Utilisez ce paramètre si vous souhaitez forcer le client à utiliser une adresse IP particulière. LOCLADDR est également utile pour forcer un canal à utiliser une adresse IPv4 ou IPv6 si un choix est disponible, ou pour utiliser un adaptateur de réseau particulier sur un système avec plusieurs adaptateurs de réseau.

La longueur maximale de LOCLADDR est MQ_LOCAL_ADDRESS_LENGTH.

Si vous omettez LOCLADDR, une adresse locale est automatiquement allouée.

adr-ip

ip-addr est une adresse réseau unique, spécifiée sous l'une des trois formes suivantes:

IPv4 en notation décimale

Par exemple 192.0.2.1

IPv6 en notation hexadécimale

Par exemple 2001:DB8:0:0:0:0:0:0

Format de nom d'hôte alphanumérique

Par exemple WWW.EXAMPLE.COM

Si une adresse IP est entrée, seul le format d'adresse est validé. L'adresse IP elle-même n'est pas validée.

Pour plus d'informations, voir [Utilisation des canaux émetteurs de cluster définis automatiquement](#) .

LONGRTY (Nombre de relances longues)


Cet attribut indique le nombre maximal de fois que le canal tente d'allouer une session à son partenaire.


L'attribut **long retry count** peut être défini de zéro à 999 999 999.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

Si la tentative d'allocation initiale échoue, le nombre de *nombre de relances abrégées* est décrémenté et le canal relance le nombre de fois restant. S'il échoue toujours, il relance un *nombre de relances longues* nombre de fois avec un intervalle de *nombre de relances longues* entre chaque tentative. S'il échoue encore, le canal se ferme. Le canal doit ensuite être redémarré avec une commande ; il n'est pas démarré automatiquement par l'initiateur de canal.

 Sous z/OS, un canal ne peut pas effectuer de nouvelle tentative si le nombre maximal de canaux (**MAXCHL**) a été dépassé.

 Sous [Multiplateformes](#), pour que la tentative soit tentée, un initiateur de canal doit être en cours d'exécution. L'initiateur de canal doit surveiller la file d'attente d'initialisation spécifiée dans la définition de la file d'attente de transmission utilisée par le canal.

Si l'initiateur de canal (sous z/OS) ou le canal (sous Multiplateformes) est arrêté alors que le canal est en cours de relance, le *nombre de relances abrégées* et le *nombre de relances longues* sont réinitialisés lorsque l'initiateur de canal ou le canal est redémarré, ou lorsqu'un message est correctement inséré sur le canal émetteur. Toutefois, si l'initiateur de canal (sous z/OS) ou le gestionnaire de files d'attente (sur Multiplatforms) est arrêté et redémarré, le *nombre de relances abrégées* et le *nombre de relances longues* ne sont pas réinitialisés. Le canal conserve les valeurs de nombre de nouvelles tentatives qu'il avait avant le redémarrage du gestionnaire de files d'attente ou le message en cours d'insertion.

Multi

Sous Multiplateformes :

1. Lorsqu'un canal passe de l'état RETRY à l'état RUNNING, le *nombre de relances abrégées* et le *nombre de relances longues* ne sont pas réinitialisés immédiatement. Ils sont réinitialisés uniquement lorsque le premier flux de messages traverse le canal avec succès après que le canal est passé à l'état RUNNING, c'est-à-dire lorsque le canal local confirme le nombre de messages envoyés à l'autre extrémité.
2. Le *nombre de relances abrégées* et le *nombre de relances longues* sont réinitialisés lorsque le canal est redémarré.

LONGTMR (intervalle entre les relances longues)

Cet attribut correspond à l'intervalle approximatif, en secondes, pendant lequel le canal doit attendre avant de tenter à nouveau d'établir une connexion, pendant le mode de relance longue.

L'intervalle entre les nouvelles tentatives peut être étendu si le canal doit attendre pour devenir actif.

Le canal tente de se connecter *nombre de relances longues* fois à cet intervalle long, après avoir essayé le nombre de *nombre de relances courtes* fois à l'intervalle des relances courtes.

Cet attribut peut être défini de zéro à 999 999.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

Attributs de canal pour les mots clés MQSC (M)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par la lettre *M*.

MAXINST (Nombre maximal d'instances)

Cet attribut indique le nombre maximal d'instances simultanées d'un canal de connexion serveur ou d'un canal AMQP pouvant être démarrés.

Nombre maximal d'instances de connexions au canal de connexion serveur

Pour un canal de connexion serveur, cet attribut indique le nombre maximal d'instances simultanées d'un canal de connexion serveur pouvant être démarrées.

Cet attribut peut être défini de zéro à 999 999 999. Une valeur de zéro indique qu'aucune connexion client n'est autorisée sur ce canal. La valeur par défaut est 999 999 999.

Si la valeur est réduite de sorte qu'elle soit inférieure au nombre d'instances du canal de connexion serveur en cours d'exécution, les canaux en cours d'exécution ne sont pas affectés. Toutefois, les nouvelles instances ne peuvent pas démarrer tant qu'un nombre suffisant d'instances existantes n'ont pas cessé de s'exécuter.

Nombre maximal d'instances de connexions de canal AMQP

Pour un canal AMQP, cet attribut indique le nombre maximal d'instances simultanées d'un canal AMQP pouvant être démarrées.

Cet attribut peut être défini de zéro à 999 999 999. Une valeur de zéro indique qu'aucune connexion client n'est autorisée sur ce canal. La valeur par défaut est 999 999 999.

Si un client tente de se connecter et que le nombre de clients connectés a atteint MAXINST, le canal ferme la connexion avec un cadre fermé. Le cadre de fermeture contient le message suivant:

```
amqp:resource-limit-exceeded
```

Si un client se connecte avec un ID déjà connecté (c'est-à-dire qu'il effectue une reprise client), la reprise aboutit, que le nombre de clients connectés ait atteint ou non MAXINST.

Pour plus d'informations, voir [Limites de canal de connexion serveur](#) .

MAXINSTC (Nombre maximal d'instances par client)

Cet attribut indique le nombre maximal d'instances simultanées d'un canal de connexion serveur pouvant être démarrées à partir d'un seul client.

Cet attribut peut être défini de zéro à 999 999 999. Une valeur de zéro indique qu'aucune connexion client n'est autorisée sur ce canal. La valeur par défaut est 999 999 999.

Si la valeur est réduite de sorte qu'elle soit inférieure au nombre d'instances du canal de connexion serveur en cours d'exécution à partir de clients individuels, les canaux en cours d'exécution ne sont pas affectés. Toutefois, les nouvelles instances de ces clients ne peuvent pas démarrer tant qu'un nombre suffisant d'instances existantes n'ont pas cessé de s'exécuter.

Cet attribut est valide uniquement pour les canaux de connexion serveur.

Pour plus d'informations, voir [Limites de canal de connexion serveur](#) .

MAXMSGL (Longueur maximale des messages)

Cet attribut indique la longueur maximale d'un message pouvant être transmis sur le canal.

Multi Sous Multiplateformes, indiquez une valeur supérieure ou égale à zéro et inférieure ou égale à la longueur maximale des messages pour le gestionnaire de files d'attente. Pour plus d'informations, voir le paramètre MAXMSGL de la commande ALTER QMGR dans [ALTER QMGR](#) .

z/OS Sous IBM MQ for z/OS, indiquez une valeur supérieure ou égale à zéro et inférieure ou égale à 104 857 600 octets (soit 100 Mo).

Etant donné que diverses implémentations de systèmes IBM MQ existent sur différentes plateformes, la taille disponible pour le traitement des messages peut être limitée dans certaines applications. Ce nombre doit refléter une taille que votre système peut gérer sans contrainte. Lorsqu'un canal démarre, le plus petit des deux nombres à chaque extrémité du canal est pris.

Remarque : Vous pouvez utiliser une taille de message maximale de 0 pour le canal, ce qui signifie que la taille doit être définie sur la valeur maximale du gestionnaire de files d'attente local.

En ajoutant la signature numérique et la clé au message, [Advanced Message Security](#) augmente la longueur du message.

Cet attribut est valide pour tous les types de canal.

MCANAME (nom de l'agent MCA)

Cet attribut est réservé. S'il est spécifié, il doit uniquement être mis à blanc et sa longueur maximale est de 20 caractères.

MCATYPE (type d'agent de canal de message)

Cet attribut peut spécifier l'agent MCA en tant que processus ou unité d'exécution.

L'exécution en tant que processus présente les avantages suivants:


- Isolement pour chaque canal offrant une plus grande intégrité
- Droits d'accès aux travaux spécifiques à chaque canal
- Contrôle de la planification des travaux


Les avantages des unités d'exécution sont les suivants:

- Utilisation très réduite du stockage
- Configuration plus facile en tapant sur la ligne de commande
- Exécution plus rapide-il est plus rapide de démarrer une unité d'exécution que de demander au système d'exploitation de démarrer un processus

Remarque : Pour les types de canal émetteur, serveur et demandeur, la valeur par défaut est `process`. Pour les types de canal émetteur de cluster et récepteur de cluster, la valeur par défaut est `thread`. Ces valeurs par défaut peuvent changer lors de votre installation.

Si vous spécifiez `process` dans la définition de canal, un processus `RUNMQCHL` est démarré. Si vous spécifiez `thread`, l'agent MCA s'exécute sur une unité d'exécution du processus `AMQRMPPA` ou du processus `RUNMQCHI` si `MQNOREMPOOL` est spécifié. Sur la machine qui reçoit les allocations entrantes, l'agent MCA s'exécute en tant qu'unité d'exécution si vous utilisez `RUNMQLSR`. Il s'exécute en tant que processus si vous utilisez `inetd`.

 Sous IBM MQ for z/OS, cet attribut est pris en charge uniquement pour les canaux avec un type de canal de récepteur de cluster.

 Sur les autres plateformes, cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

MCAUSER (ID utilisateur de l'agent MCA)

Cet attribut correspond à l'identificateur utilisateur (chaîne) à utiliser par l'agent MCA pour l'autorisation d'accès aux ressources IBM MQ .

Remarque : Une autre façon de fournir un ID utilisateur pour l'exécution d'un canal consiste à utiliser des enregistrements d'authentification de canal. Avec les enregistrements d'authentification de canal, différentes connexions peuvent utiliser le même canal tout en utilisant des données d'identification différentes. Si `MCAUSER` sur le canal est défini et que des enregistrements d'authentification de canal sont utilisés pour s'appliquer au même canal, les enregistrements d'authentification de canal sont prioritaires. La valeur `MCAUSER` de la définition de canal est utilisée uniquement si l'enregistrement d'authentification de canal utilise `USERSRC` (`CHANNEL`).

Cette autorisation inclut (si le droit `PUT` est `DEF`) l'insertion du message dans la file d'attente de destination pour les canaux récepteur ou demandeur.

Sous IBM MQ for Windows, l'identificateur utilisateur peut être qualifié de domaine à l'aide du format `user@domain`, où `domain` doit correspondre au domaine des systèmes Windows du système local ou à un domaine sécurisé.

Si cet attribut est vide, l'agent MCA utilise son ID utilisateur par défaut. Pour plus d'informations, voir [DEFINE CHANNEL](#).

Cet attribut est valide pour les types de canal suivants:

- Récepteur
- Demandeur
- Connexion serveur
- Récepteur de cluster

MODENAME (nom de mode LU 6.2)

Cet attribut doit être utilisé pour les connexions LU 6.2. Il apporte une définition supplémentaire aux caractéristiques de session de la connexion lorsqu'une allocation de session de communication est effectuée.

Lorsque vous utilisez des informations complémentaires pour les communications SNA, le nom de mode est défini dans l'objet côté communication CPI-C ou dans les informations complémentaires APPC, et cet attribut doit être laissé à blanc ; dans le cas contraire, il doit être défini sur le nom de mode SNA.

Le nom doit comporter entre un et huit caractères alphanumériques.

Cet attribut est valide uniquement pour les types de canal suivants:

- Expéditeur
- serveur
- Demandeur
- Connexion client
- Émetteur de cluster
- Récepteur de cluster

MONCHL (Surveillance)

Cet attribut contrôle la collecte des données de surveillance en ligne.

Les valeurs possibles sont les suivantes :

QMGR

La collecte des données de surveillance en ligne est héritée de la valeur de l'attribut MONCHL dans l'objet gestionnaire de files d'attente. Cette valeur est la valeur par défaut.

Non

La collecte des données de surveillance en ligne pour ce canal est désactivée.

BAS

Faible ratio de collecte de données avec un impact minimal sur les performances. Toutefois, les résultats de surveillance affichés peuvent ne pas être à jour.

MOYENNE

Un rapport modéré de la collecte de données avec un effet limité sur les performances du système.

ÉLEVÉ

Un ratio élevé de collecte de données avec la possibilité d'un effet sur les performances. Cependant, les résultats de surveillance affichés sont les plus récents.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Connexion serveur
- Émetteur de cluster

- Récepteur de cluster

Pour plus d'informations sur les données de surveillance, voir [Affichage des données de surveillance des files d'attente et des canaux](#).

MRDATA (données utilisateur de l'exit de relance de message)

son attribut spécifie les données transmises à l'exit de relance de message du canal lorsqu'il est appelé.

Cet attribut est valide pour les types de canal suivants:

- Récepteur
- Demandeur
- Récepteur de cluster

MREXIT (nom de l'exit de relance de message)

Cet attribut indique le nom du programme d'exit utilisateur à exécuter par l'exit utilisateur de relance de message.

Laissez cette zone à blanc si aucun programme d'exit de relance de message n'est actif.

Le format et la longueur maximale du nom dépendent de la plateforme, comme pour «RCVEXIT (nom de l'exit de réception)», à la page 124. Toutefois, vous ne pouvez spécifier qu'un seul exit de relance de message.

Cet attribut est valide pour les types de canal suivants:

- Récepteur
- Demandeur
- Récepteur de cluster

MRRTY (Nombre de relances de message)

Cet attribut indique le nombre de fois où le canal tente de redistribuer le message.

Cet attribut contrôle l'action de l'agent MCA uniquement si le nom de l'exit de relance de message est vide. Si le nom de l'exit n'est pas vide, la valeur de MRRTY est transmise à l'exit, mais le nombre de tentatives effectuées (le cas échéant) est contrôlé par l'exit et non par cet attribut.

La valeur doit être comprise entre 0 et 999 999 999. La valeur zéro signifie qu'aucune tentative supplémentaire n'est effectuée. La valeur par défaut est 10.

Cet attribut est valide pour les types de canal suivants:

- Récepteur
- Demandeur
- Récepteur de cluster

MRTMR (Intervalle entre les nouvelles tentatives de message)

Cet attribut indique l'intervalle de temps minimal, en millisecondes, qui doit s'écouler avant que le canal puisse relancer l'opération MQPUT.

Cet attribut contrôle l'action de l'agent MCA uniquement si le nom de l'exit de relance de message est vide. Si le nom de l'exit n'est pas vide, la valeur de MRTMR est transmise à l'exit pour être utilisée par l'exit, mais l'intervalle entre les nouvelles tentatives est contrôlé par l'exit et non par cet attribut.

La valeur doit être comprise entre 0 et 999 999 999. La valeur zéro signifie que la nouvelle tentative est effectuée dès que possible (si la valeur de MRRTY est supérieure à zéro). La valeur par défaut est 1000.

Cet attribut est valide pour les types de canal suivants:

- Récepteur
- Demandeur
- Récepteur de cluster

MSGDATA (données utilisateur d'exit de message)

Cet attribut spécifie les données utilisateur qui sont transmises aux exits de message de canal.

Vous pouvez exécuter une séquence d'exits de message. Les limitations de la longueur des données utilisateur et un exemple de spécification de MSGDATA pour plusieurs exits sont présentés pour RCVDATA. Voir «[RCVDATA \(données utilisateur de l'exit de réception\)](#)», à la page 124.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

MSGEXIT (Nom de l'exit de message)

Cet attribut indique le nom du programme d'exit utilisateur à exécuter par l'exit de message du canal.

Cet attribut peut être une liste de noms de programmes à exécuter successivement. Laissez cette zone vide si aucun exit de message de canal n'est actif.

Le format et la longueur maximale de cet attribut dépendent de la plateforme, comme pour «[RCVEXIT \(nom de l'exit de réception\)](#)», à la page 124.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

Attributs de canal pour les mots clés MQSC (N-R)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par les lettres *N* à *R*.

NETPRTY (Priorité de connexion réseau)

L'attribut de canal NETPRTY indique la priorité d'un canal CLUSRCVR . La valeur doit être comprise entre 0 et 9, où 0 correspond à la priorité la plus basse et 9 à la priorité la plus élevée.

Utilisez l'attribut NETPRTY pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Compte tenu d'un ensemble de canaux de rang égal, la mise en cluster choisit le chemin avec la priorité la plus élevée lorsque plusieurs chemins sont disponibles.

Un exemple typique d'utilisation de l'attribut de canal NETPRTY consiste à différencier les réseaux qui ont des coûts ou des vitesses différents et à connecter les mêmes destinations.

Remarque : Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).

NPMSPEED (vitesse des messages non persistants)

Cet attribut indique la vitesse à laquelle les messages non persistants doivent être envoyés.

Les valeurs possibles sont les suivantes :

NORMALE


Les messages non persistants sur un canal sont transférés dans des transactions.

RAPIDE

Les messages non persistants sur un canal ne sont pas transférés au sein des transactions.

La valeur par défaut est FAST. L'avantage est que les messages non persistants deviennent disponibles pour une extraction beaucoup plus rapide. L'inconvénient est que, comme ils ne font pas partie d'une transaction, les messages peuvent être perdus en cas d'échec de transmission ou si le canal s'arrête lorsque les messages sont en transit. Voir [Sécurité des messages](#).

Remarques :

1.  Si les journaux de reprise actifs pour IBM MQ for z/OS sont commutés et archivés plus fréquemment que prévu, étant donné que les messages envoyés via un canal sont non persistants, la définition de NPMSPEED (FAST) sur les extrémités émettrice et réceptrice du canal peut réduire le système SYSTEM.CHANNEL.SYNCQ .
2. Si vous constatez une utilisation élevée de l'unité centrale liée aux mises à jour apportées à SYSTEM.CHANNEL.SYNCQ, la définition de NPMSPEED (FAST) peut réduire considérablement l'utilisation de l'unité centrale.


Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

PASSWORD (Mot de passe)

Cet attribut indique un mot de passe qui peut être utilisé par l'agent MCA lors de la tentative d'ouverture d'une session LU 6.2 sécurisée avec un agent MCA distant.

Vous pouvez indiquer un mot de passe d'une longueur maximale de 12 caractères, bien que seuls les 10 premiers caractères soient utilisés.

 Sous IBM MQ for z/OS, cet attribut est valide uniquement pour les canaux de connexion client.

 Sur les autres plateformes, cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Demandeur
- Connexion client
- Émetteur de cluster

PORT (numéro de port)

Indiquez le numéro de port utilisé pour connecter le client AMQP.

Le port par défaut des connexions AMQP 1.0 est 5672. Si vous utilisez déjà le port 5672, vous pouvez en spécifier un autre.

PUTAUT (droit PUT)

Cet attribut indique le type de traitement de sécurité à effectuer par l'agent MCA.

Utilisez cet attribut pour choisir le type de traitement de sécurité à exécuter par l'agent MCA lors de l'exécution:

- une commande MQPUT vers la file d'attente de destination (pour les canaux de message), ou
- Un appel MQI (pour les canaux MQI).

z/OS Sous z/OS, les ID utilisateur vérifiés et le nombre d'ID utilisateur vérifiés dépendent du paramètre du profil hlq.RESLEVEL de la classe MQADMIN RACF . Selon le niveau d'accès de l'ID utilisateur de l'initiateur de canal à hlq.RESLEVEL, zéro, un ou deux ID utilisateur sont vérifiés. Pour savoir combien d'ID utilisateur sont vérifiés, voir [RESLEVEL](#) et les connexions d'initiateur de canal. Pour plus d'informations sur les ID utilisateur vérifiés, voir [ID utilisateur utilisés par l'initiateur de canal](#).

Vous pouvez choisir l'un des éléments suivants :

Sécurité des processus, également appelée autorité par défaut (DEF)

L'ID utilisateur par défaut est utilisé.

Multi Sur Multiplatforms, l'ID utilisateur utilisé pour vérifier les droits d'ouverture sur la file d'attente est celui du processus ou de l'utilisateur exécutant l'agent MCA à l'extrémité réceptrice du canal de transmission de messages.

z/OS Sous z/OS, l'ID utilisateur reçu du réseau et l'ID utilisateur dérivé de [MCAUSER](#) peuvent être utilisés, en fonction du nombre d'ID utilisateur à vérifier.

Les files d'attente sont ouvertes avec cet ID utilisateur et l'option d'ouverture MQOO_SET_ALL_CONTEXT.

Sécurité du contexte (CTX)

L'ID utilisateur issu des informations de contexte associées au message est utilisé comme ID utilisateur de remplacement.

Le *UserIdentifier* du descripteur de message est déplacé dans la zone *AlternateUserId* du descripteur d'objet. La file d'attente est ouverte avec les options d'ouverture MQOO_SET_ALL_CONTEXT et MQOO_ALTERNATE_USER_AUTHORITY.

Multi Sur Multiplatforms, l'ID utilisateur utilisé pour vérifier les droits d'ouverture sur la file d'attente pour MQOO_SET_ALL_CONTEXT et MQOO_ALTERNATE_USER_AUTHORITY est celui du processus ou de l'utilisateur exécutant l'agent MCA à l'extrémité réceptrice du canal de message. L'ID utilisateur utilisé pour vérifier les droits d'ouverture sur la file d'attente pour MQOO_OUTPUT est *UserIdentifier* dans le descripteur de message.

z/OS Sous z/OS, l'ID utilisateur reçu du réseau ou dérivé de [MCAUSER](#) peut être utilisé, ainsi que l'ID utilisateur provenant des informations de contexte dans le descripteur de message, en fonction du nombre d'ID utilisateur à vérifier.

La sécurité du contexte (CTX) n'est pas prise en charge sur les canaux de connexion serveur.

z/OS Sécurité de l'agent MCA uniquement (ONLYMCA)

L'ID utilisateur dérivé de [MCAUSER](#) est utilisé.

Les files d'attente sont ouvertes avec l'option d'ouverture MQOO_SET_ALL_CONTEXT.

Cette valeur s'applique uniquement à z/OS.




z/OS Sécurité de l'agent ALTMCA (Alternate Message Channel Agent)

L'ID utilisateur provenant des informations de contexte (zone *UserIdentifier*) du descripteur de message peut être utilisé, ainsi que l'ID utilisateur dérivé de MCAUSER, en fonction du nombre d'ID utilisateur à vérifier.


Cette valeur s'applique uniquement à z/OS.

Pour plus de détails sur les zones de contexte et les options d'ouverture, voir Contrôle des informations de contexte.

Pour plus d'informations sur la sécurité, voir:

- Sécurisation
-  Configuration de la sécurité sur AIX, Linux, and Windows
-  Configuration de la sécurité sur IBM i
-  Configuration de la sécurité sur z/OS

Cet attribut est valide pour les types de canal suivants:

- Récepteur
- Demandeur
-  Connexion serveur (z/OS uniquement)
- Récepteur de cluster

QMNAME (nom du gestionnaire de files d'attente)

son attribut spécifie le nom du gestionnaire de files d'attente ou du groupe de gestionnaires de files d'attente auquel une application IBM MQ MQI client peut demander une connexion.

Cet attribut est valide pour les types de canal suivants:

- Connexion client

QSGDISP (disposition)



Cet attribut indique la disposition du canal dans un groupe de partage de files d'attente. Elle est valide sous z/OS uniquement.

Les valeurs sont les suivantes :

QMGR

Le canal est défini dans l'ensemble de pages du gestionnaire de files d'attente qui exécute la commande. Cette valeur est la valeur par défaut.

GROUPE

Le canal est défini dans le référentiel partagé. Cette valeur est autorisée uniquement s'il existe un environnement de gestionnaire de files d'attente partagées. Lorsqu'un canal est défini avec QSGDISP (GROUP), la commande DEFINE CHANNEL (name) NOREPLACE QSGDISP (COPY) est générée automatiquement et envoyée à tous les gestionnaires de files d'attente actifs pour qu'ils effectuent des copies locales sur l'ensemble de pages 0. Pour les gestionnaires de files d'attente qui ne sont pas actifs ou qui rejoignent le groupe de partage de files d'attente à une date ultérieure, la commande est générée au démarrage du gestionnaire de files d'attente.

COPIER

Le canal est défini dans l'ensemble de pages du gestionnaire de files d'attente qui exécute la commande, en copiant sa définition à partir du canal QSGDISP (GROUP) du même nom. Cette valeur est autorisée uniquement s'il existe un environnement de gestionnaire de files d'attente partagées.

Cet attribut est valide pour tous les types de canal.

RCVDATA (données utilisateur de l'exit de réception)

Cet attribut spécifie les données utilisateur transmises à l'exit de réception.

Vous pouvez exécuter une séquence d'exits de réception. La chaîne de données utilisateur d'une série d'exits doit être séparée par une virgule, des espaces ou les deux. Exemple :

```
RCVDATA(exit1_data exit2_data)
MSGDATA(exit1_data,exit2_data)
SENDDATA(exit1_data, exit2_data)
```

ALW Sur les systèmes IBM MQ for UNIX et Windows , la longueur de la chaîne de noms d'exit et des chaînes de données utilisateur est limitée à 500 caractères.

IBM i Dans IBM MQ for IBM i, vous pouvez spécifier jusqu'à 10 noms d'exit et la longueur des données utilisateur pour chacun d'eux est limitée à 32 caractères.

z/OS Dans IBM MQ for z/OS, vous pouvez spécifier jusqu'à huit chaînes de données utilisateur d'une longueur de 32 caractères chacune.

Cet attribut est valide pour tous les types de canal.

RCVEXIT (nom de l'exit de réception)

Cet attribut indique le nom du programme d'exit utilisateur à exécuter par l'exit utilisateur de réception de canal.

Cet attribut peut être une liste de noms de programmes à exécuter successivement. Laissez cette zone vide si aucun exit utilisateur de réception de canal n'est actif.

Le format et la longueur maximale de cet attribut dépendent de la plateforme:

- z/OS** Sous z/OS , il s'agit d'un nom de module de chargement d'une longueur maximale de 8 caractères, à l'exception des canaux de connexion client dont la longueur maximale est de 128 caractères.
- IBM i** Sous IBM i, il se présente sous la forme suivante:

```
libname/programe
```

lorsqu'il est spécifié dans les commandes CL.

Lorsqu'il est spécifié dans les commandes IBM MQ (MQSC), il se présente sous la forme suivante:

```
programe libname
```

où *programe* occupe les 10 premiers caractères et *libname* les 10 seconds caractères (les deux espaces étant complétés à droite si nécessaire). La longueur maximale de la chaîne est de 20 caractères.

- Linux** **AIX** Sous AIX and Linux, il se présente sous la forme suivante:

```
libraryname(functionname)
```

La longueur maximale de la chaîne est de 40 caractères.

- **Windows** Sous Windows, il se présente sous la forme suivante:

```
dllname(functionname)
```

où *dllname* est spécifié sans le suffixe .DLL. La longueur maximale de la chaîne est de 40 caractères.

- **z/OS** Lors de la définition automatique du canal émetteur de cluster sur z/OS, les noms d'exit de canal sont convertis au format z/OS . Si vous souhaitez contrôler la conversion des noms d'exit, vous pouvez écrire un exit de définition automatique de canal. Pour plus d'informations, voir [Programme d'exit de définition automatique de canal](#).

Vous pouvez indiquer une liste de noms de programme d'exit de réception, d'envoi ou de message. Les noms doivent être séparés par une virgule, un espace ou les deux. Exemple :

```
RCVEXIT(exit1 exit2)  
MSGEXIT(exit1,exit2)  
SENDEXIT(exit1, exit2)
```

La longueur totale de la chaîne de noms d'exit et des chaînes de données utilisateur pour un type particulier d'exit est limitée à 500 caractères.

- **IBM i** Sous IBM MQ for IBM i, vous pouvez répertorier jusqu'à 10 noms d'exit.
- **z/OS** Sous IBM MQ for z/OS, vous pouvez répertorier jusqu'à huit noms d'exit.

Cet attribut est valide pour tous les types de canal.

Attributs de canal pour les mots clés MQSC (S)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par la lettre S.

SCYDATA (données utilisateur de l'exit de sécurité)

Cet attribut spécifie les données utilisateur qui sont transmises à l'exit de sécurité.

La longueur maximale est de 32 caractères.

Cet attribut est valide pour tous les types de canal.

SCYEXIT (nom de l'exit de sécurité)

Cet attribut indique le nom du programme d'exit à exécuter par l'exit de sécurité de canal.

Laissez cette zone vide si aucun exit de sécurité de canal n'est actif.

Le format et la longueur maximale du nom dépendent de la plateforme, comme pour [RCVEXIT](#). Toutefois, vous ne pouvez spécifier qu'un seul exit de sécurité.

Pour plus d'informations sur les exits de sécurité, voir [Programmes d'exit de sécurité de canal](#).

Cet attribut est valide pour tous les types de canal.

SENDDATA (Envoi de données utilisateur d'exit)

Cet attribut spécifie les données utilisateur qui sont transmises à l'exit d'émission.

Vous pouvez exécuter une séquence d'exits d'envoi. Les limitations relatives à la longueur des données utilisateur et un exemple de spécification de SENDDATA pour plusieurs exits sont affichées pour RCVDATA. Voir [RCVDATA](#).

Cet attribut est valide pour tous les types de canal.

SENDEXIT (nom de l'exit d'envoi)

Cet attribut indique le nom du programme d'exit à exécuter par l'exit d'émission de canal.

Cet attribut peut être une liste de noms de programmes à exécuter en séquence. Laissez cette zone vide si aucun exit d'émission de canal n'est actif.

Le format et la longueur maximale de cet attribut dépendent de la plateforme, comme pour [RCVEXIT](#).

Cet attribut est valide pour tous les types de canal.

SEQWRAP (encapsulation de numéro de séquence)

Cet attribut indique le numéro le plus élevé que le numéro de séquence de message atteint avant de redémarrer à 1.

La valeur du nombre doit être suffisamment élevée pour éviter qu'un nombre ne soit réémis alors qu'il est encore utilisé par un message antérieur. Les deux extrémités d'un canal doivent avoir la même valeur de bouclage de numéro de séquence lorsqu'un canal démarre ; sinon, une erreur se produit.

La valeur peut être comprise entre 100 et 999 999 999.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

SHORTRTY (Nombre de relances abrégées)

Cet attribut indique le nombre maximal de fois que le canal tente d'allouer une session à son partenaire.

L'attribut SHORTRTY peut être défini de zéro à 999 999 999.

Si plusieurs adresses IP ont été définies dans le canal et qu'une reconnexion est nécessaire, IBM MQ évalue la définition de canal et tente de se connecter à chaque adresse IP dans l'ordre dans lequel elle est définie jusqu'à ce qu'une connexion soit établie ou que toutes les adresses aient été tentées.

Dans ce cas, SHORTRTY est lié au nombre total de tentatives de reconnexion du canal global et non aux adresses IP individuelles.

Si la tentative d'allocation initiale échoue, le *nombre de tentatives abrégées* est décrémenté et le canal relance le nombre de tentatives restantes avec un intervalle, défini dans l'attribut **short retry interval**, entre chaque tentative. S'il échoue encore, il relance *nombre de relances longues* fois avec un intervalle de *nombre de relances longues* entre chaque tentative. S'il échoue encore, le canal se ferme.

z/OS Sous z/OS, un canal ne peut pas effectuer de nouvelle tentative si le nombre maximal de canaux (**MAXCHL**) a été dépassé.

Multi Sous [Multiplateformes](#), pour que la tentative soit tentée, un initiateur de canal doit être en cours d'exécution. L'initiateur de canal doit surveiller la file d'attente d'initialisation spécifiée dans la définition de la file d'attente de transmission utilisée par le canal.

Si l'initiateur de canal (sous z/OS) ou le canal (sous [Multiplateformes](#)) est arrêté lors de la relance du canal, le *nombre de relances abrégées* et le *nombre de relances longues* sont réinitialisés lorsque l'initiateur de canal ou le canal est redémarré, ou lorsqu'un message est correctement inséré sur le canal émetteur. Toutefois, si l'initiateur de canal (sous z/OS) ou le gestionnaire de files d'attente (sur [Multiplatforms](#)) est arrêté et redémarré, le *nombre de relances abrégées* et le *nombre de relances longues*

ne sont pas réinitialisés. Le canal conserve les valeurs de nombre de nouvelles tentatives qu'il avait avant le redémarrage du gestionnaire de files d'attente ou le message en cours d'insertion.

Multi

Sous Multiplateformes :

1. Lorsqu'un canal passe de l'état `RETRY` à l'état `RUNNING`, le *nombre de relances abrégées* et le *nombre de relances longues* ne sont pas réinitialisés immédiatement. Ils sont réinitialisés uniquement lorsque le premier flux de messages traverse le canal avec succès après que le canal est passé à l'état `RUNNING`, c'est-à-dire lorsque le canal local confirme le nombre de messages envoyés à l'autre extrémité.
2. Le *nombre de relances abrégées* et le *nombre de relances longues* sont réinitialisés lorsque le canal est redémarré.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

SHORTTMR (Intervalle entre les nouvelles tentatives)

Cet attribut indique l'intervalle approximatif, en secondes, pendant lequel le canal doit attendre avant de tenter à nouveau d'établir une connexion, pendant le mode de relance courte.

L'intervalle entre les nouvelles tentatives peut être étendu si le canal doit attendre pour devenir actif.

Cet attribut peut être défini de zéro à 999 999.

Si plusieurs adresses IP ont été définies dans le canal et qu'une reconnexion est nécessaire, IBM MQ évalue la définition de canal et tente de se connecter à chaque adresse IP dans l'ordre dans lequel elle est définie jusqu'à ce qu'une connexion soit établie ou que toutes les adresses aient été tentées.

Dans ce cas, `SHORTTMR` indique combien de temps le canal global attend pour redémarrer le processus de connexion, et non les adresses IP individuelles.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Émetteur de cluster
- Récepteur de cluster

SPLPROT (Protection des règles de sécurité)

z/OS

Cet attribut indique comment un agent Message Channel de serveur à serveur doit gérer la protection des messages lorsque `AMS` est actif et qu'une règle applicable existe.

Cet attribut peut être défini sur:

PASSTHRU

Sur les canaux émetteur, serveur, récepteur et demandeur

REMOVE

Sur les canaux émetteur et serveur

ASPOLICY

Sur les canaux récepteur et demandeur

Cet attribut est valide pour les types de canal suivants:

- Expéditeur

- serveur
- Récepteur
- Demandeur

SSLCAUTH (authentification de client SSL)

L'attribut **SSLCAUTH** indique si le canal doit recevoir et authentifier un certificat TLS d'un client TLS.

SSLCAUTH est un attribut facultatif. Les valeurs possibles de cet attribut sont les suivantes:

Facultatif

Si le client TLS homologue envoie un certificat, le certificat est traité normalement, mais l'authentification n'échoue pas si aucun certificat n'est envoyé.

REQUIRED

Si le client TLS n'envoie pas de certificat, l'authentification échoue.

La valeur par défaut est REQUIRED.

Vous pouvez spécifier une valeur pour **SSLCAUTH** sur une définition de canal non TLS. Il s'agit d'une définition de canal sur laquelle l'attribut **SSLCIPH** est manquant ou vide.

Pour plus d'informations sur SSLCAUTH, voir [DEFINE CHANNEL \(MQTT\)](#) et [Sécurisation](#).

L'attribut **SSLCAUTH** est valide sur tous les types de canal pouvant recevoir un flux d'initiation de canal, à l'exception des canaux émetteurs. Cet attribut est valide pour les types de canal suivants:

- serveur
- Récepteur
- Demandeur
- Connexion serveur
- Récepteur de cluster

SSLCIPH (spécification de chiffrement SSL)

L'attribut **SSLCIPH** spécifie un alias ou un nom unique CipherSpec pour une connexion TLS.

Chaque définition de canal IBM MQ inclut l'attribut **SSLCIPH**. La valeur est une chaîne d'une longueur maximale de 32 caractères.

L'attribut **SSLCIPH** est valide uniquement pour les canaux dont le type de transport (**TRPTYPE**) est TCP. Si **TRPTYPE** n'est pas TCP, les données sont ignorées et aucun message d'erreur n'est émis.

Remarques :

- L'attribut **SSLCIPH** peut contenir une valeur vide, ce qui signifie que vous n'utilisez pas TLS. Si une extrémité du canal a un attribut **SSLCIPH** vide, l'autre extrémité du canal doit également avoir un attribut SSLCIPH vide.

► Multi

Si l'option [SecureCommsOnly](#) est activée, la communication en texte en clair n'est pas prise en charge et le canal ne démarre pas.

- Sinon, si **SSLCIPH** contient une valeur non vide, la valeur peut être un alias ou un CipherSpec nommé. Les canaux négocient le CipherSpec le plus puissant pris en charge par les deux extrémités du canal.
- Un client .NET entièrement géré peut spécifier la valeur spéciale *NÉGOCIER. Cette option permet au canal de sélectionner la version de protocole la plus récente prise en charge par l'infrastructure .NET et de négocier un CipherSpec pris en charge par le serveur.

L'attribut **SSLCIPH** est valide uniquement pour les canaux dont le type de transport (**TRPTYPE**) est TCP. Si **TRPTYPE** n'est pas TCP, les données sont ignorées et aucun message d'erreur n'est émis.

Pour plus d'informations sur **SSLCIPH**, voir [DEFINE CHANNEL](#) et [Spécification de CipherSpecs](#)

SSLPEER (homologue SSL)

L'attribut **SSLPEER** permet de vérifier le nom distinctif (DN) du certificat à partir du gestionnaire de files d'attente ou du client homologue à l'autre extrémité d'un canal IBM MQ .

Remarque : Une autre façon de restreindre les connexions dans les canaux en les comparant au nom distinctif du sujet TLS consiste à utiliser des enregistrements d'authentification de canal. Avec les enregistrements d'authentification de canal, différents modèles de nom distinctif de sujet TLS peuvent être appliqués au même canal. Si **SSLPEER** sur le canal et un enregistrement d'authentification de canal sont utilisés pour s'appliquer au même canal, le certificat entrant doit correspondre aux deux modèles pour la connexion.

Si le nom distinctif reçu de l'homologue ne correspond pas à la valeur **SSLPEER** , le canal ne démarre pas.

SSLPEER est un attribut facultatif. Si aucune valeur n'est spécifiée, le nom distinctif de l'homologue n'est pas vérifié lorsque le canal est démarré.

La longueur maximale de l'attribut **SSLPEER** dépend de la plateforme:

- **z/OS** Sous z/OS, la longueur maximale de l'attribut est de 256 octets.
- **Multi** Sur toutes les autres plateformes, il est de 1024 octets.

Les enregistrements d'authentification de canal offrent une plus grande flexibilité lors de l'utilisation de **SSLPEER** et prennent en charge une longueur maximale de 1024 octets sur toutes les plateformes.

La vérification des valeurs d'attribut **SSLPEER** dépend également de la plateforme:

- **z/OS** Sous z/OS, les valeurs d'attribut utilisées ne sont pas vérifiées. Si vous entrez des valeurs incorrectes, le canal échoue au démarrage et des messages d'erreur sont consignés dans le journal des erreurs aux deux extrémités du canal. Un événement Erreur SSL de canal est également généré aux deux extrémités du canal.
- **Multi** Sous Multiplateformes qui prend en charge **SSLPEER**, la validité de la chaîne est vérifiée lors de sa première saisie.

Vous pouvez spécifier une valeur pour **SSLPEER** sur une définition de canal non TLS, sur laquelle l'attribut SSLCIPH est manquant ou vide. Vous pouvez l'utiliser pour désactiver temporairement TLS pour le débogage sans avoir à effacer et à entrer à nouveau les paramètres TLS ultérieurement.

L'attribut **SSLPEER** est valide pour tous les types de canal.

Pour plus d'informations sur l'utilisation de **SSLPEER**, voir [SET CHLAUTH](#), [Sécurisation](#) et [Enregistrements d'authentification de canal](#).

STATCHL (statistiques de canal)

Cet attribut contrôle la collecte des données statistiques pour les canaux.

Les valeurs possibles sont les suivantes:

QMGR

La collecte de données statistiques pour ce canal est basée sur la définition de l'attribut de gestionnaire de files d'attente STATCHL. Cette valeur est la valeur par défaut.

Non

La collecte de données statistiques pour ce canal est désactivée.

BAS

La collecte de données statistiques pour ce canal est activée avec un faible taux de collecte de données.

MOYENNE

La collecte de données statistiques pour ce canal est activée avec un rapport moyen de collecte de données.

ÉLEVÉ

La collecte de données statistiques pour ce canal est activée avec un taux élevé de collecte de données.

Pour plus d'informations sur les statistiques de canal, voir [Référence de surveillance](#).

z/OS Sur les systèmes z/OS, l'activation de ce paramètre active simplement la collecte de données statistiques, quelle que soit la valeur que vous sélectionnez. La spécification de LOW, MEDIUM ou HIGH n'a pas d'impact sur vos résultats. Vous devez activer ce paramètre pour pouvoir collecter des enregistrements de comptabilité de canal.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Récepteur
- Demandeur
- Émetteur de cluster
- Récepteur de cluster

Attributs de canal pour les mots clés MQSC (T-Z)

Liste alphabétique des attributs de canal pour les mots clés MQSC, commençant par les lettres T à Z.

TPNAME (nom de programme de transaction LU 6.2)

Cet attribut doit être utilisé pour les connexions LU 6.2. Il s'agit du nom, ou du nom générique, du programme de transaction (MCA) à exécuter à l'extrémité éloignée de la liaison.

Lorsque vous utilisez des informations complémentaires pour les communications SNA, le nom du programme de transaction est défini dans l'objet côté communication CPI-C ou dans les informations complémentaires APPC et cet attribut doit être laissé à blanc. Sinon, ce nom est requis par les canaux émetteurs et demandeurs.

Le nom peut atteindre 64 caractères.

Le nom doit être défini sur le nom du programme de transaction SNA, sauf si CONNAME contient un nom d'objet secondaire, auquel cas il doit être mis à blanc. Le nom réel est pris à la place de l'objet CPI-C Communications Side Object ou du fichier d'informations APPC.

Ces informations sont définies de différentes manières sur différentes plateformes ; voir [Configuration de la mise en file d'attente répartie](#) pour plus d'informations sur la configuration de la communication pour votre plateforme.

Cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Demandeur
- Connexion client
- Émetteur de cluster
- Récepteur de cluster

TPROOT (Racine de rubrique)

Cet attribut spécifie la racine de rubrique d'un canal AMQP.

Vous pouvez utiliser l'attribut TPROOT pour spécifier une racine de rubrique pour un canal AMQP. L'utilisation de cet attribut garantit qu'une application MQ Light, lorsqu'elle est déployée dans un

gestionnaire de files d'attente, ne publie pas de messages dans les zones de l'arborescence de rubriques utilisées par d'autres applications ou ne s'y abonne pas.

La valeur par défaut de TPROOT est SYSTEM.BASE.TOPIC. Avec cette valeur, la chaîne de rubrique qu'un client AMQP utilise pour la publication ou l'abonnement n'a pas de préfixe et le client peut échanger des messages avec d'autres applications pub/sub MQ . Pour que les clients AMQP publient et s'abonnent sous un préfixe de rubrique, créez d'abord un objet de rubrique MQ avec une chaîne de rubrique définie sur le préfixe de votre choix, puis remplacez la valeur de l'attribut TPROOT du canal AMQP par le nom de l'objet de rubrique MQ que vous avez créé. L'exemple suivant montre la racine de rubrique définie sur APPGROUP1.BASE.TOPIC pour le canal AMQP MYAMQP:

```
DEFINE CHANNEL(MYAMQP) CHLTYPE(AMQP) TPROOT(APPGROUP1.BASE.TOPIC) PORT(5673)
```

Remarque : Si la valeur de l'attribut TPROOT ou la chaîne de rubrique qui la sous-tend est modifiée, les rubriques AMQP existantes et leurs messages risquent d'être orphelines.

TRPTYPE (Type de transport)

Cet attribut indique le type de transport à utiliser.

Les valeurs possibles sont les suivantes:

Tableau 58. Options de type de transport	
Valeur	Type de transport
LU62	LU 6.2
TCP	TCP/IP
NETBIOS	NetBIOS «1», à la page 131
SPX	SPX (Sequenced Packet Exchange protocol) «1», à la page 131

Remarques :

1. A utiliser sous Windows. Peut également être utilisé sur z/OS pour définir des canaux de connexion client à utiliser sur Windows.

Cet attribut est valide pour tous les types de canal, mais il est ignoré par les agents MCA.

USECLTID (Utiliser l'ID client)

Indiquez si l'ID client est utilisé pour la connexion sur un canal AMQP. Définissez la valeur sur Yes ou No.

USEDLQ (Utiliser la file d'attente des messages non livrés)

Cet attribut détermine si la file d'attente de rebut (ou la file d'attente de messages non livrés) est utilisée lorsque les messages ne peuvent pas être distribués par les canaux.

Les valeurs possibles sont les suivantes :

NO

Les messages qui ne peuvent pas être distribués par un canal sont traités comme un échec. Le canal supprime ces messages, ou les extrémités du canal, conformément à la valeur de NPMSPEED.

YES (valeur par défaut)

Si l'attribut DEADQ du gestionnaire de files d'attente fournit le nom d'une file d'attente de rebut, il est utilisé, sinon le comportement est NO.

USERID (ID utilisateur)

Cet attribut indique l'ID utilisateur à utiliser par l'agent MCA lors de la tentative d'ouverture d'une session SNA sécurisée avec un agent MCA éloigné.

Vous pouvez spécifier un identificateur utilisateur de tâche de 20 caractères.

À la réception, si les mots de passe sont conservés au format chiffré et que le logiciel LU 6.2 utilise une méthode de chiffrement différente, une tentative de démarrage du canal échoue avec des détails de sécurité non valides. Vous pouvez éviter cet incident en modifiant la configuration SNA de réception de l'une des manières suivantes:

- Désactivez la substitution de mot de passe ou
- Définissez un ID utilisateur et un mot de passe de sécurité.

z/OS Sous IBM MQ for z/OS, cet attribut est valide uniquement pour les canaux de connexion client.

Multi Sur les autres plateformes, cet attribut est valide pour les types de canal suivants:

- Expéditeur
- serveur
- Demandeur
- Connexion client
- Émetteur de cluster

XMITQ (nom de la file d'attente de transmission)

Cet attribut indique le nom de la file d'attente de transmission à partir de laquelle les messages sont extraits.

Indiquez le nom de la file d'attente de transmission à associer à ce canal émetteur ou serveur, qui correspond au gestionnaire de files d'attente du côté éloigné du canal. Vous pouvez attribuer à la file d'attente de transmission le même nom que le gestionnaire de files d'attente à l'extrémité éloignée.

Cet attribut est obligatoire pour les canaux de type émetteur ou serveur et n'est pas valide pour les autres types de canal.

Commandes et attributs de cluster IBM MQ

Il existe des commandes de cluster MQSC et PCF que vous pouvez utiliser pour actualiser ou réinitialiser un cluster, ou pour afficher, reprendre ou interrompre un gestionnaire de files d'attente de cluster. En outre, les commandes MQSC et PCF qui définissent les canaux, les files d'attente et les gestionnaires de files d'attente possèdent des attributs qui s'appliquent aux clusters. Certains de ces attributs sont utilisés par l'algorithme de gestion de la charge de travail du cluster.

Commandes MQSC

Les commandes MQSC sont affichées car elles sont entrées par l'administrateur système sur la console de commande. N'oubliez pas que vous n'avez pas besoin d'émettre les commandes de cette manière. Il existe un certain nombre d'autres méthodes, en fonction de votre plateforme, par exemple:

- **IBM i** Sous IBM MQ for IBM i, vous exécutez des commandes MQSC de manière interactive à partir de l'option 26 de **WRKMQM**. Vous pouvez également utiliser des commandes CL ou stocker des commandes MQSC dans un fichier et utiliser la commande CL **STRMQMMQSC**.
- **z/OS** Sous z/OS, vous pouvez utiliser la fonction COMMAND de l'utilitaire **CSQUTIL**, les panneaux d'opérations et de contrôle ou la console z/OS.

- Sur toutes les autres plateformes, vous pouvez stocker les commandes dans un fichier et utiliser **runmqsc**.

Dans une commande MQSC, un nom de cluster spécifié à l'aide de l'attribut CLUSTER peut comporter jusqu'à 48 caractères.

Une liste de noms de cluster spécifiée à l'aide de l'attribut CLUSNL peut contenir jusqu'à 256 noms. Pour créer une liste de noms de cluster, utilisez la commande DEFINE NAMELIST .

IBM MQ Explorer

L'interface graphique d' IBM MQ Explorer peut administrer un cluster avec des gestionnaires de files d'attente de référentiel sous IBM WebSphere MQ for z/OS 6.0 ou version ultérieure. Il n'est pas nécessaire de désigner un référentiel supplémentaire sur un système distinct. Pour les versions antérieures d' IBM MQ for z/OS, IBM MQ Explorer ne peut pas administrer un cluster avec des gestionnaires de files d'attente de référentiel. Vous devez donc désigner un référentiel supplémentaire sur un système que le IBM MQ Explorer peut administrer.

Sous IBM MQ for Windows et IBM MQ for Linux, vous pouvez également utiliser IBM MQ Explorer pour utiliser des clusters. Vous pouvez également utiliser le client IBM MQ Explorer autonome.

A l'aide de l' IBM MQ Explorer, vous pouvez afficher les files d'attente de cluster et vous renseigner sur le statut des canaux émetteur et récepteur de cluster. IBM MQ Explorer inclut deux assistants que vous pouvez utiliser pour vous guider dans les tâches suivantes:

- Créer un cluster
- Joindre un gestionnaire de files d'attente indépendant à un cluster

Équivalents PCF des commandes MQSC spécifiquement pour l'utilisation des clusters

Commande MQSC	Commande PCF équivalente
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

Information associée

[Mise en cluster : meilleures pratiques d'utilisation REFRESH CLUSTER](#)

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Les commandes DEFINE CHANNEL, ALTER CHANNEL et DISPLAY CHANNEL possèdent deux paramètres CHLTYPE spécifiques pour les clusters: CLUSRCVR et CLUSSDR. Pour définir un canal récepteur de cluster, utilisez la commande DEFINE CHANNEL en spécifiant CHLTYPE (CLUSRCVR) . De nombreux attributs d'une définition de canal récepteur de cluster sont identiques à ceux d'une définition de canal récepteur ou émetteur. Pour définir un canal émetteur de cluster, utilisez la commande DEFINE CHANNEL , en spécifiant CHLTYPE (CLUSSDR) , et la plupart des attributs que vous utilisez pour définir un canal émetteur.

Il n'est plus nécessaire de spécifier le nom du gestionnaire de files d'attente de référentiel complet lorsque vous définissez un canal émetteur de cluster. Si vous connaissez la convention de dénomination utilisée pour les canaux de votre cluster, vous pouvez créer une définition CLUSSDR à l'aide de la construction +QMNAME+ . La construction +QMNAME+ n'est pas prise en charge sous z/OS. Après la

connexion, IBM MQ modifie le nom du canal et remplace le nom de gestionnaire de files d'attente de référentiel complet correct par +QMNAME+. Le nom de canal résultant est tronqué à 20 caractères.

Pour plus d'informations sur les conventions de dénomination, voir [Conventions de dénomination de cluster](#).

Cette technique fonctionne uniquement si votre convention de dénomination des canaux inclut le nom du gestionnaire de files d'attente. Par exemple, vous définissez un gestionnaire de files d'attente de référentiel complet appelé QM1 dans un cluster appelé CLUSTER1 avec un canal récepteur de cluster appelé CLUSTER1.QM1.ALPHA. Tous les autres gestionnaires de files d'attente peuvent définir un canal émetteur de cluster pour ce gestionnaire de files d'attente à l'aide du nom de canal, CLUSTER1.+QMNAME+.ALPHA.

Si vous utilisez la même convention de dénomination pour tous vos canaux, sachez qu'une seule définition +QMNAME+ peut exister à la fois.

Les attributs suivants sur les commandes DEFINE CHANNEL et ALTER CHANNEL sont spécifiques aux canaux de cluster:

Grappe

L'attribut CLUSTER indique le nom du cluster auquel ce canal est associé. Vous pouvez également utiliser l'attribut CLUSNL.

CLUSNL

L'attribut CLUSNL spécifie une liste de noms de noms de cluster.

NETPRTY

Récepteurs de cluster uniquement.

L'attribut NETPRTY indique une priorité de réseau pour le canal. NETPRTY aide les routines de gestion de charge de travail. S'il existe plusieurs routes possibles vers une destination, la routine de gestion de la charge de travail sélectionne celle dont la priorité est la plus élevée.

CLWLPRTY

Le paramètre CLWLPRTY applique un facteur de priorité aux canaux de la même destination à des fins de gestion de la charge de travail. Ce paramètre indique la priorité du canal à des fins de distribution de la charge de travail du cluster. La valeur doit être comprise entre zéro et 9, où zéro correspond à la priorité la plus basse et 9 à la priorité la plus élevée.

CLWLRANK

Le paramètre CLWLRANK applique un facteur de classement à un canal à des fins de gestion de charge de travail. Ce paramètre indique le rang d'un canal à des fins de distribution de la charge de travail du cluster. La valeur doit être comprise entre zéro et 9, où zéro correspond au rang le plus bas et 9 au rang le plus élevé.

CLWLWGHT

Le paramètre CLWLWGHT applique un facteur de pondération à un canal à des fins de gestion de la charge de travail. CLWLWGHT pondère le canal de sorte que la proportion de messages envoyés via ce canal puisse être contrôlée. L'algorithme de charge de travail de cluster utilise CLWLWGHT pour biaiser le choix de destination afin qu'un plus grand nombre de messages puissent être envoyés sur un canal particulier. Par défaut, tous les attributs de pondération de canal ont la même valeur par défaut. L'attribut de pondération vous permet d'allouer un canal sur une machine UNIX puissante dont le poids est supérieur à celui d'un autre canal sur un petit PC de bureau. Une pondération plus élevée signifie que l'algorithme de charge de travail du cluster sélectionne la machine UNIX plus fréquemment que le PC comme destination des messages.

CONNAME

La valeur CONNAME spécifiée dans une définition de canal récepteur de cluster est utilisée dans l'ensemble du cluster pour identifier l'adresse réseau du gestionnaire de files d'attente. Prenez soin de sélectionner une valeur pour le paramètre CONNAME qui est résolue dans votre cluster IBM MQ. N'utilisez pas de nom générique. N'oubliez pas que la valeur spécifiée sur le canal récepteur de cluster est prioritaire sur toute valeur spécifiée dans un canal émetteur de cluster correspondant.

Ces attributs de la commande DEFINE CHANNEL et de la commande ALTER CHANNEL s'appliquent également à la commande DISPLAY CHANNEL.

Remarque : Les canaux émetteurs de cluster définis automatiquement tirent leurs attributs de la définition de canal récepteur de cluster correspondante dans le gestionnaire de files d'attente de réception. Même s'il existe un canal émetteur de cluster défini manuellement, ses attributs sont automatiquement modifiés pour s'assurer qu'ils correspondent aux attributs de la définition de récepteur de cluster correspondante. Prenez en compte que vous pouvez, par exemple, définir un CLUSRCVR sans spécifier de numéro de port dans le paramètre CONNAME , tout en définissant manuellement un CLUSSDR qui spécifie un numéro de port. Lorsque le CLUSSDR défini automatiquement remplace le numéro défini manuellement, le numéro de port (extrait de CLUSRCVR) devient vide. Le numéro de port par défaut est utilisé et le canal échoue.

Remarque : La commande DISPLAY CHANNEL n'affiche pas les canaux définis automatiquement. Toutefois, vous pouvez utiliser la commande DISPLAY CLUSQMGR pour examiner les attributs des canaux émetteurs de cluster définis automatiquement.

La commande DISPLAY CHSTATUS permet d'afficher l'état d'un canal émetteur ou récepteur de cluster. Cette commande indique le statut des canaux définis manuellement et des canaux définis automatiquement.

Les fichiers PCF équivalents sont MQCMD_CHANGE_CHANNEL, MQCMD_COPY_CHANNEL, MQCMD_CREATE_CHANNEL et MQCMD_INQUIRE_CHANNEL.

Omission de la valeur CONNAME dans une définition CLUSRCVR

Dans certains cas, vous pouvez omettre la valeur CONNAME dans une définition CLUSRCVR . Vous ne devez pas omettre la valeur CONNAME sous z/OS.

Multi Sur [Multiplateformes](#), le paramètre de nom de connexion TCP/IP d'un canal récepteur de cluster est facultatif. Si vous laissez le nom de connexion vide, IBM MQ génère un nom de connexion automatiquement en utilisant le port par défaut et l'adresse IP actuelle du système. Vous pouvez remplacer le numéro de port par défaut, et continuer à utiliser l'adresse IP actuelle du système. Pour chaque nom de connexion laissez le nom IP à blanc, puis indiquez le numéro de port entre parenthèses, par exemple :

(1415)

Le fichier **CONNAME** généré est toujours dans le format décimal à points (IPv4) ou hexadécimal (IPv6), plutôt que sous la forme d'un nom d'hôte DNS alphanumérique.

Cette fonction est utile lorsque vous disposez de machines utilisant le protocole DHCP (Dynamic Host Configuration Protocol). Si vous n'indiquez pas de valeur pour CONNAME sur un canal CLUSRCVR , vous n'avez pas besoin de modifier la définition CLUSRCVR . DHCP vous attribue une nouvelle adresse IP.

Si vous indiquez un blanc pour CONNAME dans la définition CLUSRCVR , IBM MQ génère un CONNAME à partir de l'adresse IP du système. Seul le CONNAME généré est stocké dans les référentiels. Les autres gestionnaires de files d'attente du cluster ne savent pas que CONNAME était à l'origine vide.

Si vous exécutez la commande DISPLAY CLUSQMGR , le CONNAME généré s'affiche. Toutefois, si vous exécutez la commande DISPLAY CHANNEL à partir du gestionnaire de files d'attente local, vous constatez que CONNAME est vide.

Si le gestionnaire de files d'attente est arrêté et redémarré avec une adresse IP différente, en raison du protocole DHCP, IBM MQ régénère le CONNAME et met à jour les référentiels en conséquence.

Concepts associés

[Equilibrage de charge dans les clusters](#)

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande `DISPLAY CLUSQMGR` permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Emettez la commande `REFRESH CLUSTER` à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande `SUSPEND QMGR` et `RESUME QMGR` pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

«Attributs de canal d'équilibrage de charge de cluster», à la page 148

Liste alphabétique des attributs de canal utilisés dans l'équilibrage de charge de cluster.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Les commandes **DEFINE QLOCAL**, **DEFINE QREMOTE** et **DEFINE QALIAS**

Les attributs de cluster des commandes `DEFINE QLOCAL`, `DEFINE QREMOTE` et `DEFINE QALIAS`, ainsi que les trois commandes `ALTER` équivalentes, sont les suivants:

Grappe

Indique le nom du cluster auquel appartient la file d'attente.

CLUSNL

Indique une liste de noms de cluster.

DEFBIND

Indique la liaison à utiliser lorsqu'une application spécifie `MQOO_BIND_AS_Q_DEF` sur l'appel `MQOPEN`. Les options de cet attribut sont les suivantes:

- Indiquez `DEFBIND(OPEN)` pour lier le descripteur de file d'attente à une instance spécifique de la file d'attente de cluster lorsque la file d'attente est ouverte. `DEFBIND(OPEN)` est la valeur par défaut de cet attribut.
- Spécifiez `DEFBIND(NOTFIXED)` pour que l'identificateur de file d'attente ne soit lié à aucune instance de la file d'attente de cluster.
- Spécifiez `DEFBIND(GROUP)` pour permettre à une application de demander qu'un groupe de messages soit alloué à la même instance de destination.

Lorsque plusieurs files d'attente portant le même nom sont annoncées dans un cluster de gestionnaires de files d'attente, les applications peuvent choisir d'envoyer tous les messages de cette application à une seule instance (`MQOO_BIND_ON_OPEN`), afin de permettre à l'algorithme de gestion de charge de travail de sélectionner la destination la plus appropriée par message

(MQOO_BIND_NOT_FIXED), ou d'autoriser une application à demander qu'un groupe de messages soit alloué à la même instance de destination (MQOO_BIND_ON_GROUP). L'équilibrage de la charge de travail est réacheminé entre les groupes de messages (sans nécessiter de MQCLOSE et de MQOPEN de la file d'attente).

Lorsque vous spécifiez DEFBIND dans une définition de file d'attente, la file d'attente est définie avec l'un des attributs MQBND_BIND_ON_OPEN, MQBND_BIND_NOT_FIXED ou MQBND_BIND_ON_GROUP. MQBND_BIND_ON_OPEN ou MQBND_BIND_ON_GROUP doit être spécifié lors de l'utilisation de groupes avec des clusters.

Vous devez définir l'attribut DEFBIND sur la même valeur sur toutes les instances de la même file d'attente de cluster.

CLWLRANK

Applique un facteur de classement à une file d'attente à des fins de gestion de la charge de travail. Le paramètre CLWLRANK n'est pas pris en charge sur les files d'attente modèles. L'algorithme de charge de travail du cluster sélectionne une file d'attente de destination avec le rang le plus élevé. Par défaut, CLWLRANK pour toutes les files d'attente est défini sur zéro.

Si la destination finale est un gestionnaire de files d'attente sur un autre cluster, vous pouvez définir le rang des gestionnaires de files d'attente de passerelle intermédiaires à l'intersection des clusters voisins. Les gestionnaires de files d'attente intermédiaires étant classés, l'algorithme de charge de travail de cluster sélectionne correctement un gestionnaire de files d'attente de destination proche de la destination finale.

La même logique s'applique aux files d'attente alias. La sélection de rang est effectuée avant la vérification de l'état du canal et, par conséquent, même les gestionnaires de files d'attente non accessibles sont disponibles pour la sélection. Cela a pour effet de permettre à un message d'être acheminé via un réseau, plutôt que de le faire sélectionner entre deux destinations possibles (comme le ferait la priorité). Ainsi, si un canal n'est pas démarré à l'endroit où le rang a été indiqué, le message n'est pas acheminé vers le rang suivant le plus élevé, mais attend qu'un canal soit disponible pour cette destination (le message est conservé dans la file d'attente de transmission).

CLWLPRTY

Applique un facteur de priorité à une file d'attente à des fins de gestion de charge de travail. L'algorithme de charge de travail du cluster sélectionne une file d'attente de destination avec la priorité la plus élevée. Par défaut, la priorité de toutes les files d'attente est définie sur zéro.

S'il existe deux files d'attente de destination possibles, vous pouvez utiliser cet attribut pour effectuer une reprise en ligne d'une destination vers l'autre destination. La sélection de la priorité est effectuée après la vérification de l'état du canal. Tous les messages sont envoyés à la file d'attente avec la priorité la plus élevée, sauf si le statut du canal vers cette destination n'est pas aussi favorable que celui des canaux vers d'autres destinations. Cela signifie que seules les destinations les plus accessibles peuvent être sélectionnées. Cela a pour effet d'établir des priorités entre plusieurs destinations qui sont toutes disponibles.

CLWLUSEQ

Indique le comportement d'une opération MQPUT pour une file d'attente. Ce paramètre indique le comportement d'une opération MQPUT lorsque la file d'attente cible comporte une instance locale et au moins une instance de cluster distant (sauf si MQPUT provient d'un canal de cluster). Ce paramètre n'est valide que pour les files d'attente locales.

Les valeurs possibles sont: QMGR (le comportement est celui spécifié par le paramètre CLWLUSEQ de la définition de gestionnaire de files d'attente), ANY (le gestionnaire de files d'attente traite la file d'attente locale comme une autre instance de la file d'attente de cluster, à des fins de distribution de la charge de travail), LOCAL (la file d'attente locale est la seule cible de l'opération MQPUT, à condition que la file d'attente locale soit activée). Le comportement de MQPUT dépend de l'algorithme de gestion de charge de travail de cluster.

Les commandes DISPLAY QUEUE et DISPLAY QCLUSTER

Les attributs des commandes DEFINE QLOCAL, DEFINE QREMOTE et DEFINE QALIAS s'appliquent également à la commande DISPLAY QUEUE.

Pour afficher des informations sur les files d'attente de cluster, indiquez le type de file d'attente QCLUSTER ou le mot clé CLUSINFO dans la commande DISPLAY QUEUE ou utilisez la commande DISPLAY QCLUSTER.

La commande DISPLAY QUEUE ou DISPLAY QCLUSTER renvoie le nom du gestionnaire de files d'attente qui héberge la file d'attente (ou les noms de tous les gestionnaires de files d'attente s'il existe plusieurs instances de la file d'attente). Elle renvoie également le nom de système de chaque gestionnaire de files d'attente qui héberge la file d'attente, le type de file d'attente représenté, ainsi que la date et l'heure auxquelles la définition est devenue disponible pour le gestionnaire de files d'attente local. Ces informations sont renvoyées à l'aide des attributs CLUSQMGR, QMID, CLUSQT, CLUSDATE et CLUSTIME .

Le nom de système du gestionnaire de files d'attente (QMID) est un nom unique généré par le système pour le gestionnaire de files d'attente.

Vous pouvez définir une file d'attente de cluster qui est également une file d'attente partagée. Par exemple, sous z/OS , vous pouvez définir:

```
DEFINE QLOCAL(MYQUEUE) CLUSTER(MYCLUSTER) QSGDISP(SHARED) CFSTRUCT(STRUCTURE)
```

Les fichiers PCF équivalents sont MQCMD_CHANGE_Q, MQCMD_COPY_Q, MQCMD_CREATE_Q et MQCMD_INQUIRE_Q.

Concepts associés

Équilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande DISPLAY CLUSQMGR permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Emettez la commande REFRESH CLUSTER à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande SUSPEND QMGR et RESUME QMGR pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

«Équilibrage de charge de cluster-Attributs de file d'attente», à la page 150

Liste alphabétique des attributs de file d'attente utilisés dans l'équilibrage de charge de cluster.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

Pour indiquer qu'un gestionnaire de files d'attente contient un référentiel complet pour un cluster, utilisez la commande **ALTER QMGR** en spécifiant l'attribut `REPOS(clustername)`. Pour spécifier une liste de plusieurs noms de cluster, définissez une liste de noms de cluster, puis spécifiez l'attribut `REPOSNL(namelist)` dans la commande **ALTER QMGR** :

```
DEFINE NAMELIST(CLUSTERLIST)
  DESCR('List of clusters whose repositories I host')
  NAMES(CLUS1, CLUS2, CLUS3)
ALTER QMGR REPOSNL(CLUSTERLIST)
```

Vous pouvez fournir des attributs de cluster supplémentaires dans la commande **ALTER QMGR**

CLWLEXIT(*name*)

Indique le nom d'un exit utilisateur à appeler lorsqu'un message est inséré dans une file d'attente de cluster.

CLWLDATA(*data*)

Indique les données à transmettre à l'exit utilisateur de charge de travail de cluster.

CLWLLEN(*length*)

Indique la quantité maximale de données de message à transmettre à l'exit utilisateur de la charge de travail du cluster

CLWLMRUC(*channels*)

Indique le nombre maximal de canaux de cluster sortants.

CLWLMRUC est un attribut de gestionnaire de files d'attente local qui n'est pas propagé autour du cluster. Il est mis à la disposition des exits de charge de travail de cluster et de l'algorithme de charge de travail de cluster qui choisit la destination des messages.

CLWLUSEQ(LOCAL|ANY)

Indique le comportement de MQPUT lorsque la file d'attente cible comporte à la fois une instance locale et au moins une instance de cluster distant. Si l'insertion émane d'un canal de cluster, cet attribut ne s'applique pas. Il est possible de spécifier CLWLUSEQ à la fois comme attribut de file d'attente et comme attribut de gestionnaire de files d'attente.

Si vous spécifiez ANY, la file d'attente locale et les files d'attente éloignées sont toutes deux des cibles possibles de MQPUT.

Si vous spécifiez LOCAL, la file d'attente locale est la seule cible de MQPUT.

Les fichiers PCF équivalents sont MQCMD_CHANGE_Q_MGR et MQCMD_INQUIRE_Q_MGR.

Concepts associés

Equilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

DISPLAY CLUSQMGR

La commande `DISPLAY CLUSQMGR` permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Émettez la commande `REFRESH CLUSTER` à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande `SUSPEND QMGR` et `RESUME QMGR` pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

«Équilibrage de charge de cluster-Attributs de gestionnaire de files d'attente», à la page 152

Liste alphabétique des attributs de gestionnaire de files d'attente utilisés dans l'équilibrage de charge de cluster.

DISPLAY CLUSQMGR

La commande `DISPLAY CLUSQMGR` permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Si vous exécutez cette commande à partir d'un gestionnaire de files d'attente doté d'un référentiel complet, les informations renvoyées s'appliquent à tous les gestionnaires de files d'attente du cluster. Sinon, les informations renvoyées s'appliquent uniquement aux gestionnaires de files d'attente qui l'intéressent. C'est-à-dire tous les gestionnaires de files d'attente à qui la commande a essayé d'envoyer un message et tous les gestionnaires de files d'attente qui sont dotés d'un référentiel complet.

Les informations incluent la plupart des attributs de canal qui s'appliquent aux canaux émetteurs et récepteurs de cluster. En outre, les attributs suivants peuvent être affichés:

Canal

Nom du canal récepteur de cluster pour le gestionnaire de files d'attente.

DATE CLUSDATE

Date à laquelle la définition est devenue disponible pour le gestionnaire de files d'attente local.

Grappe

Clusters dans lesquels se trouve le gestionnaire de files d'attente.

HEURE_CLEE

Date à laquelle la définition est devenue disponible pour le gestionnaire de files d'attente local.

DEFTYPE

Mode de définition du gestionnaire de files d'attente. `DEFTYPE` peut être l'une des valeurs suivantes:

CLUSDR

Un canal émetteur de cluster a été défini administrativement sur le gestionnaire de files d'attente local mais n'a pas encore été reconnu par le gestionnaire de files d'attente cible. Pour être dans cet état, le gestionnaire de files d'attente local a défini un canal émetteur de cluster manuel, mais le gestionnaire de files d'attente récepteur n'a pas accepté les informations du cluster. Cela peut être dû au fait que le canal n'a jamais été établi en raison de la disponibilité ou d'une erreur dans la configuration de l'émetteur de cluster, par exemple une non-concordance dans la propriété `CLUSTER` entre les définitions de l'émetteur et du récepteur. Il s'agit d'une condition transitoire ou d'un état d'erreur qui doit être examiné.

CLUSDRA

Cette valeur représente un gestionnaire de files d'attente de cluster reconnu automatiquement, aucun canal émetteur de cluster n'est défini en local. Il s'agit du type par défaut des gestionnaires

de files d'attente de cluster pour lesquels le gestionnaire de files d'attente local n'a pas de configuration locale mais en a été informé. Par exemple

- Si le gestionnaire de files d'attente local est un gestionnaire de files d'attente de référentiel complet, il doit correspondre à la valeur DEFTYPE pour tous les gestionnaires de files d'attente de référentiel partiel du cluster.
- Si le gestionnaire de files d'attente local est un référentiel partiel, il peut s'agir de l'hôte d'une file d'attente de cluster qui est utilisée à partir de ce gestionnaire de files d'attente local ou d'un second gestionnaire de files d'attente de référentiel complet avec lequel ce gestionnaire de files d'attente a été informé de son utilisation.

Si la valeur de DEFTYPE est CLUSSDRA et que les gestionnaires de files d'attente locaux et distants sont tous deux des référentiels complets pour le cluster nommé, la configuration n'est pas correcte car un canal émetteur de cluster défini en local doit être défini pour la convertir en un DEFTYPE de CLUSDRB.

CLUSDRB

Un canal émetteur de cluster a été défini administrativement sur le gestionnaire de files d'attente local et accepté comme canal de cluster valide par le gestionnaire de files d'attente cible. Il s'agit du type DEFTYPE attendu d'un gestionnaire de files d'attente de référentiel partiel configuré manuellement pour le gestionnaire de files d'attente de référentiel complet. Il doit également s'agir du type par défaut de tout CLUSQMGR d'un référentiel complet à un autre référentiel complet dans le cluster. Les canaux émetteurs de cluster manuels ne doivent pas être configurés dans des référentiels partiels ou à partir d'un gestionnaire de files d'attente de référentiel partiel vers plusieurs référentiels complets. Si un DEFTYPE de CLUSSDRB est vu dans l'une de ces situations, il doit être examiné et corrigé.

CLUSRCVR

Défini administrativement en tant que canal récepteur de cluster sur le gestionnaire de files d'attente local. Représente le gestionnaire de files d'attente local dans le cluster.

Remarque : Pour identifier les gestionnaires de files d'attente de référentiel CLUSQGRs qui sont des gestionnaires de files d'attente de référentiel complet pour le cluster, voir la propriété QMTYPE .

Pour plus d'informations sur la définition des canaux de cluster, voir Canaux de cluster.

QMTYPE

Indique s'il contient un référentiel complet ou seulement un référentiel partiel.

Statut

Statut du canal émetteur de cluster pour ce gestionnaire de files d'attente.

SUSPEND

Indique si le gestionnaire de files d'attente est suspendu.

version

Version de l'installation IBM MQ à laquelle le gestionnaire de files d'attente de cluster est associé.

La version a le format VVRRMMFF :

- VV : Version
- RR : édition
- MM : niveau de maintenance
- FF : niveau de correctif

XMITQ

File d'attente de transmission du cluster utilisée par le gestionnaire de files d'attente.

Voir aussi la commande DISPLAY QCLUSTER . Ceci est brièvement décrit dans DISPLAY QUEUE et dans la section DISPLAY QUEUE and DISPLAY QCLUSTER commands de «Attributs de cluster disponibles dans les commandes de définition de file d'attente», à la page 136. Pour des exemples d'utilisation de DISPLAY QCLUSTER, recherchez "DISPLAY QCLUSTER" et "DIS QCLUSTER" dans l'ensemble d'informations.

Concepts associés

Equilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

Actualiser le cluster

Emettez la commande REFRESH CLUSTER à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande SUSPEND QMGR et RESUME QMGR pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

Commande MQSC **DISPLAY CLUSQMGR**

Actualiser le cluster

Emettez la commande REFRESH CLUSTER à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

Il existe trois formes de cette commande:

REFRESH CLUSTER(clustername) REPOS(NO)

La valeur par défaut Le gestionnaire de files d'attente conserve la connaissance de tous les gestionnaires de files d'attente de cluster définis en local et des files d'attente de cluster et de tous les gestionnaires de files d'attente de cluster qui sont des référentiels complets. En outre, si le gestionnaire de files d'attente est un référentiel complet pour le cluster, il conserve également la connaissance des autres gestionnaires de files d'attente du cluster. Tout le reste est supprimé de la copie locale du référentiel et régénéré à partir des autres référentiels complets du cluster. Les canaux de cluster ne sont pas arrêtés si REPOS(NO) est utilisé. Un référentiel complet utilise ses canaux CLUSSDR pour informer le reste du cluster qu'il a terminé son actualisation.

REFRESH CLUSTER(clustername) REPOS(YES)

Outre le comportement par défaut, les objets représentant des gestionnaires de files d'attente de cluster de référentiel complet sont également actualisés. Il n'est pas possible d'utiliser cette option si le gestionnaire de files d'attente est un référentiel complet. Si la commande est utilisée, elle échouera avec l'erreur AMQ9406/CSQX406E consignée. S'il s'agit d'un référentiel complet, vous devez d'abord le modifier afin qu'il ne soit pas un référentiel complet pour le cluster en question. L'emplacement de référentiel complet est récupéré à partir des définitions CLUSSDR définies manuellement. Une fois

que l'actualisation avec REPOS (YES) a été émise, le gestionnaire de files d'attente peut être modifié de sorte qu'il soit à nouveau un référentiel complet, si nécessaire.

REFRESH CLUSTER(*)

Régénère le gestionnaire de files d'attente dans tous les clusters dont il est membre. Si cette option est utilisée avec REPOS (YES) REFRESH CLUSTER (*), elle a pour effet supplémentaire de forcer le gestionnaire de files d'attente à redémarrer sa recherche de référentiels complets à partir des informations figurant dans les définitions CLUSSDR locales. La recherche a lieu même si le canal CLUSSDR connecte le gestionnaire de files d'attente à plusieurs clusters.

Remarque : Pour les grands clusters, l'utilisation de la commande **REFRESH CLUSTER** peut affecter le fonctionnement du cluster et à nouveau tous les 27 jours lorsque les objets de cluster envoient automatiquement les mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster.

Concepts associés

Equilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande **DISPLAY CLUSQMGR** permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande **SUSPEND QMGR** et **RESUME QMGR** pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

Information associée

Mise en cluster : meilleures pratiques d'utilisation REFRESH CLUSTER

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

Vous pouvez exécuter la commande **RESET CLUSTER** uniquement à partir des gestionnaires de files d'attente de référentiel complet. La commande prend deux formes, selon que vous référencez le gestionnaire de files d'attente par nom ou par identificateur.

1.

```
RESET CLUSTER( clustername
) QMNAME( qmname ) ACTION(FORCEREMOVE) QUEUES(NO)
```

2.

```
RESET CLUSTER( clustername
) QMID( qmid ) ACTION(FORCEREMOVE) QUEUES(NO)
```

Vous ne pouvez pas spécifier à la fois QMNAME et QMID. Si vous utilisez QMNAME et qu'il existe plusieurs gestionnaires de files d'attente dans le cluster avec ce nom, la commande n'est pas exécutée. Utilisez QMID à la place de QMNAME pour vous assurer que la commande **RESET CLUSTER** est exécutée.

La spécification de QUEUES (NO) sur une commande **RESET CLUSTER** est la valeur par défaut. La spécification de QUEUES (YES) supprime du cluster les références aux files d'attente de cluster appartenant au gestionnaire de files d'attente. Les références sont supprimées en plus de la suppression du gestionnaire de files d'attente du cluster lui-même.

Les références sont supprimées même si le gestionnaire de files d'attente de cluster n'est pas visible dans le cluster ; peut-être parce qu'il a été précédemment supprimé de force, sans l'option QUEUES .

Vous pouvez utiliser la commande **RESET CLUSTER** si, par exemple, un gestionnaire de files d'attente a été supprimé mais que des canaux récepteurs de cluster sont toujours définis sur le cluster. Au lieu d'attendre que IBM MQ supprime ces définitions (ce qu'il fait automatiquement), vous pouvez exécuter la commande **RESET CLUSTER** pour effectuer une opération de rangement plus tôt. Tous les autres gestionnaires de files d'attente du cluster sont alors informés que le gestionnaire de files d'attente n'est plus disponible.

Si un gestionnaire de files d'attente est temporairement endommagé, vous pouvez le signaler aux autres gestionnaires de files d'attente du cluster avant qu'ils ne tentent de l'envoyer. **RESET CLUSTER** supprime le gestionnaire de files d'attente endommagé. Ensuite, lorsque le gestionnaire de files d'attente endommagé fonctionne à nouveau, utilisez la commande **REFRESH CLUSTER** pour inverser l'effet de **RESET CLUSTER** et renvoyer le gestionnaire de files d'attente au cluster. Si le gestionnaire de files d'attente se trouve dans un cluster de publication/abonnement, vous devez rétablir tout abonnement de proxy requis. Voir les [remarques sur REFRESH CLUSTER pour les clusters de publication/abonnement](#).

Remarque : Pour les grands clusters, l'utilisation de la commande **REFRESH CLUSTER** peut affecter le fonctionnement du cluster et à nouveau tous les 27 jours lorsque les objets de cluster envoient automatiquement les mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir [L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster](#).

L'utilisation de la commande **RESET CLUSTER** est le seul moyen de supprimer les canaux émetteurs de cluster définis automatiquement.

Important : Si le canal défini automatiquement à supprimer est en attente de validation, **RESET CLUSTER** ne supprime pas immédiatement ce canal. Dans cette situation, vous devez émettre une commande [RESOLVE CHANNEL](#) avant la commande **RESET CLUSTER** .

Il est peu probable que vous ayez besoin de cette commande dans des circonstances normales. Le support IBM peut vous conseiller d'émettre la commande pour nettoyer les informations de cluster détenues par les gestionnaires de files d'attente de cluster. N'utilisez pas cette commande comme raccourci pour supprimer un gestionnaire de files d'attente d'un cluster. La méthode appropriée pour supprimer un gestionnaire de files d'attente d'un cluster est décrite dans la rubrique [Suppression d'un gestionnaire de files d'attente d'un cluster](#).

Etant donné que les référentiels ne conservent les informations que pendant 90 jours, un gestionnaire de files d'attente qui a été supprimé de force peut se reconnecter à un cluster. Il se reconnecte automatiquement, sauf s'il a été supprimé. Si vous souhaitez empêcher un gestionnaire de files d'attente de rejoindre un cluster, vous devez prendre les mesures de sécurité appropriées.

Toutes les commandes de cluster, à l'exception de **DISPLAY CLUSQMGR**, fonctionnent de manière asynchrone. Les commandes qui modifient les attributs d'objet impliquant la mise en cluster mettent à jour l'objet et envoient une demande au processeur de référentiel. La syntaxe des commandes d'utilisation des clusters est vérifiée et une demande est envoyée au processeur de référentiel.

Les demandes envoyées au processeur de référentiel sont traitées de manière asynchrone, avec les demandes de cluster reçues d'autres membres du cluster. Le traitement peut prendre un temps considérable s'ils doivent être propagés dans l'ensemble du cluster pour déterminer s'ils ont abouti ou non.

Concepts associés

Équilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande `DISPLAY CLUSQMGR` permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Émettez la commande `REFRESH CLUSTER` à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande `SUSPEND QMGR` et `RESUME QMGR` pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

RESET CLUSTER (réinitialisation d'un cluster)

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande `SUSPEND QMGR` et `RESUME QMGR` pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

Lorsqu'un gestionnaire de files d'attente est suspendu à partir d'un cluster, il ne reçoit pas de messages sur les files d'attente de cluster qu'il héberge s'il existe une file d'attente disponible du même nom sur un autre gestionnaire de files d'attente du cluster. Toutefois, les messages qui sont explicitement ciblés sur ce gestionnaire de files d'attente, ou dans lesquels la file d'attente cible est disponible uniquement sur ce gestionnaire de files d'attente, sont toujours dirigés vers ce gestionnaire de files d'attente.

La réception de messages entrants supplémentaires lorsque le gestionnaire de files d'attente est suspendu peut être évitée en arrêtant les canaux récepteurs de cluster pour ce cluster. Pour arrêter les canaux récepteurs d'un cluster, utilisez le mode `FORCE` de la commande `SUSPEND QMGR`.

Concepts associés

Equilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

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.

Tâches associées

Gestion d'un gestionnaire de files d'attente

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande **DISPLAY CLUSQMGR** permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Emettez la commande **REFRESH CLUSTER** à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR

RESUME QMGR

Equilibrage de charge dans les clusters

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

Les destinations appropriées sont choisies, par l'algorithme de gestion de la charge de travail du cluster, en fonction de la disponibilité du gestionnaire de files d'attente et de la file d'attente et d'un certain nombre d'attributs spécifiques à la charge de travail du cluster associés aux canaux, aux files d'attente et aux gestionnaires de files d'attente. Ces attributs sont décrits dans les sous-rubriques.

Après avoir configuré les attributs spécifiques à la charge de travail du cluster, si la configuration ne se comporte pas comme prévu, examinez les détails de la manière dont l'algorithme choisit un gestionnaire de files d'attente. Voir «Algorithme de gestion de la charge de travail du cluster», à la page 152. Si les résultats de cet algorithme ne répondent pas à vos besoins, vous pouvez écrire un programme d'exit utilisateur de charge de travail de cluster et utiliser cet exit pour acheminer les messages vers la file d'attente de votre choix dans le cluster. Voir «Ecriture et compilation des exits de charge de travail de cluster».

Tableau 59. Récapitulatif des attributs spécifiques à la charge de travail du cluster

Nom de l'attribut	Description
Attributs de canal	
<u>CLWLPRTY</u>	Indique l'ordre de priorité des canaux pour la distribution de la charge de travail du cluster.
<u>CLWLRANK</u>	Indique le rang des canaux pour la distribution de la charge de travail du cluster.
<u>CLWLWGHT</u>	Indique le poids appliqué aux canaux CLUSSDR et CLUSRCVR pour la distribution de la charge de travail du cluster.
<u>NETPRTY</u>	Indique la priorité d'un canal CLUSRCVR .
Attributs File d'attente	
<u>CLWLPRTY</u>	Indique la priorité des files d'attente locales, distantes ou d'alias pour la distribution de la charge de travail du cluster.
<u>CLWLRANK</u>	Indique le rang d'une file d'attente locale, éloignée ou alias pour la distribution de la charge de travail du cluster.
<u>CLWLUSEQ</u>	Indique si une instance locale d'une file d'attente est choisie comme destination de préférence à d'autres instances dans un cluster.
Attributs de gestionnaire de files d'attente	
<u>CLWLMRUC</u>	Définit le nombre de canaux les plus récemment choisis.
<u>CLWLUSEQ</u>	Indique si une instance locale d'une file d'attente est privilégiée en tant que destination par rapport à d'autres instances de la file d'attente dans un cluster.

Concepts associés

 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.

Référence associée

Attributs de cluster disponibles dans les commandes de définition de canal

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Attributs de cluster disponibles dans les commandes de définition de file d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande DISPLAY CLUSQMGR permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Émettez la commande `REFRESH CLUSTER` à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande `SUSPEND QMGR` et `RESUME QMGR` pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

Attributs de canal d'équilibrage de charge de cluster

Liste alphabétique des attributs de canal utilisés dans l'équilibrage de charge de cluster.

Remarque : Spécifiez les attributs de canal de charge de travail de cluster sur les canaux récepteurs de cluster au niveau des gestionnaires de files d'attente cible. Tout équilibrage que vous spécifiez sur les canaux émetteurs de cluster correspondants est susceptible d'être ignoré. Voir [Canaux de cluster](#).

CLWLPRTY (Priorité de charge de travail de cluster)

L'attribut de canal `CLWLPRTY` indique l'ordre de priorité des canaux pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 0 et 9, où 0 correspond à la priorité la plus basse et 9 à la priorité la plus élevée.

Utilisez l'attribut de canal `CLWLPRTY` pour définir un ordre de priorité pour les destinations de cluster disponibles. IBM MQ sélectionne les destinations ayant la priorité la plus élevée avant de sélectionner les destinations ayant la priorité de destination de cluster la plus faible. S'il existe plusieurs destinations avec la même priorité, elle sélectionne la destination la moins récemment utilisée.

S'il existe deux destinations possibles, vous pouvez utiliser cet attribut pour autoriser la reprise en ligne. Les messages sont envoyés au gestionnaire de files d'attente avec le canal dont la priorité est la plus élevée. S'il devient indisponible, les messages passent au gestionnaire de files d'attente de priorité supérieure suivant. Les gestionnaires de files d'attente de priorité inférieure agissent comme des réserves.

IBM MQ vérifie le statut des canaux avant de les hiérarchiser. Seuls les gestionnaires de files d'attente disponibles sont candidats à la sélection.

Remarques :

- Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).
- La disponibilité d'un gestionnaire de files d'attente éloignées dépend du statut du canal de ce gestionnaire de files d'attente. Lorsque les canaux démarrent, leur état change plusieurs fois, certains d'entre eux étant moins préférentiels à l'algorithme de gestion de la charge de travail du cluster. En pratique, cela signifie que les destinations de priorité inférieure (sauvegarde) peuvent être choisies alors que les canaux vers les destinations de priorité supérieure (principale) sont en cours de démarrage.
- Si vous devez vous assurer qu'aucun message n'est acheminé vers une destination de sauvegarde, n'utilisez pas `CLWLPRTY`. Envisagez d'utiliser des files d'attente distinctes ou `CLWLRANK` avec un basculement manuel du serveur principal vers le serveur de sauvegarde.

CLWLRANK (rang de charge de travail de cluster)

L'attribut de canal **CLWLRANK** indique le rang des canaux pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 0 et 9, où 0 correspond au rang le plus bas et 9 au rang le plus élevé.

Utilisez l'attribut de canal **CLWLRANK** si vous souhaitez contrôler la destination finale des messages envoyés à un gestionnaire de files d'attente dans un autre cluster. Contrôlez le choix de la destination finale en définissant le rang des canaux qui connectent un gestionnaire de files d'attente aux gestionnaires de files d'attente de passerelle à l'intersection des clusters.

Lorsque vous définissez **CLWLRANK**, les messages empruntent une route spécifique via les clusters interconnectés vers une destination de rang supérieur. Par exemple, les messages parviennent à un gestionnaire de files d'attente de passerelle qui peut les envoyer à l'un des deux gestionnaires de files d'attente à l'aide des canaux classés 1 et 2. Ils sont envoyés automatiquement au gestionnaire de files d'attente connecté par un canal ayant le rang le plus élevé, dans ce cas le canal est envoyé au gestionnaire de files d'attente classé 2.

IBM MQ obtient le rang des canaux avant de vérifier le statut des canaux. L'obtention du rang avant la vérification de l'état du canal signifie que même les canaux non accessibles sont disponibles pour la sélection. Il permet aux messages d'être acheminés via le réseau même si la destination finale n'est pas disponible.

Remarques :

- Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).
- Si vous avez également utilisé l'attribut de priorité **CLWLPRTY**, IBM MQ effectue une sélection entre les destinations disponibles. Si un canal n'est pas disponible pour la destination dont le rang est le plus élevé, le message est conservé dans la file d'attente de transmission. Il est libéré lorsque le canal devient disponible. Le message n'est pas envoyé à la destination disponible suivante dans l'ordre de classement.

CLWLWGHT (pondération de charge de travail du cluster)

L'attribut de canal CLWLWGHT indique le poids appliqué aux canaux CLUSSDR et CLUSRCVR pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 1 et 99, où 1 correspond à la pondération la plus faible et 99 à la pondération la plus élevée.

Utilisez CLWLWGHT pour envoyer des serveurs avec plus de messages de puissance de traitement. Plus le poids du canal est élevé, plus le nombre de messages envoyés via ce canal est élevé.

Remarques :

- Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).
- Lorsque CLWLWGHT est modifié à partir de la valeur par défaut de 50 sur n'importe quel canal, l'équilibrage de la charge de travail dépend du nombre total de fois où chaque canal a été choisi pour un message envoyé à n'importe quelle file d'attente en cluster. Pour plus d'informations, voir [«Algorithme de gestion de la charge de travail du cluster»](#), à la page 152.

NETPRTY (Priorité de connexion réseau)

L'attribut de canal NETPRTY indique la priorité d'un canal CLUSRCVR . La valeur doit être comprise entre 0 et 9, où 0 correspond à la priorité la plus basse et 9 à la priorité la plus élevée.

Utilisez l'attribut NETPRTY pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Compte tenu d'un ensemble de canaux de rang égal, la mise en cluster choisit le chemin avec la priorité la plus élevée lorsque plusieurs chemins sont disponibles.

Un exemple typique d'utilisation de l'attribut de canal NETPRTY consiste à différencier les réseaux qui ont des coûts ou des vitesses différents et à connecter les mêmes destinations.

Remarque : Indiquez cet attribut sur le canal récepteur de cluster au niveau du gestionnaire de files d'attente cible. Tout équilibrage que vous spécifiez sur le canal émetteur de cluster correspondant est susceptible d'être ignoré. Voir [Canaux de cluster](#).

Concepts associés

Algorithme de gestion de la charge de travail du cluster

L'algorithme de gestion de la charge de travail utilise des attributs d'équilibrage de charge et de nombreuses règles pour sélectionner la destination finale des messages placés dans les files d'attente de cluster.

Référence associée

Équilibrage de charge de cluster-Attributs de file d'attente

Liste alphabétique des attributs de file d'attente utilisés dans l'équilibrage de charge de cluster.

Équilibrage de charge de cluster-Attributs de gestionnaire de files d'attente

Liste alphabétique des attributs de gestionnaire de files d'attente utilisés dans l'équilibrage de charge de cluster.

«Attributs de cluster disponibles dans les commandes de définition de canal», à la page 133

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

Équilibrage de charge de cluster-Attributs de file d'attente

Liste alphabétique des attributs de file d'attente utilisés dans l'équilibrage de charge de cluster.

CLWLPRTY

L'attribut de file d'attente **CLWLPRTY** indique la priorité des files d'attente locales, distantes ou alias pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 0 et 9, où 0 correspond à la priorité la plus basse et 9 à la priorité la plus élevée.

Utilisez l'attribut de file d'attente **CLWLPRTY** pour définir une préférence pour les files d'attente de destination. IBM MQ sélectionne les destinations ayant la priorité la plus élevée avant de sélectionner les destinations ayant la priorité de destination de cluster la plus faible. S'il existe plusieurs destinations avec la même priorité, elle sélectionne la destination la moins récemment utilisée.

IBM MQ obtient la priorité des gestionnaires de files d'attente après avoir vérifié le statut du canal. Seuls les gestionnaires de files d'attente disponibles sont candidats à la sélection.

Remarque :

La disponibilité d'un gestionnaire de files d'attente éloignées dépend du statut du canal de ce gestionnaire de files d'attente. Lorsque les canaux démarrent, leur état change plusieurs fois, certains d'entre eux étant moins préférentiels à l'algorithme de gestion de la charge de travail du cluster. En pratique, cela signifie que les destinations de priorité inférieure (sauvegarde) peuvent être choisies alors que les canaux vers les destinations de priorité supérieure (principale) sont en cours de démarrage.

Si vous devez vous assurer qu'aucun message n'est acheminé vers une destination de sauvegarde, n'utilisez pas **CLWLPRTY**. Envisagez d'utiliser des files d'attente distinctes ou **CLWLRANK** avec un basculement manuel de la base de données principale vers la base de données de sauvegarde.

S'il existe deux destinations possibles, vous pouvez utiliser cet attribut pour autoriser la reprise en ligne. Le gestionnaire de files d'attente dont la priorité est la plus élevée reçoit des demandes et les gestionnaires de files d'attente dont la priorité est la plus faible servent de réserves. Si le gestionnaire de files d'attente dont la priorité est la plus élevée échoue, le gestionnaire de files d'attente dont la priorité est la plus élevée qui est disponible prend le relais.

CLWLRANK

L'attribut de file d'attente **CLWLRANK** indique le rang d'une file d'attente locale, éloignée ou alias pour la distribution de la charge de travail du cluster. La valeur doit être comprise entre 0 et 9, où 0 correspond au rang le plus bas et 9 au rang le plus élevé.

Utilisez l'attribut de file d'attente **CLWLRANK** si vous souhaitez contrôler la destination finale des messages envoyés à un gestionnaire de files d'attente dans un autre cluster. Lorsque vous définissez **CLWLRANK**, les messages empruntent une route spécifique via les clusters interconnectés vers une destination de rang supérieur.

Par exemple, vous pouvez avoir défini deux gestionnaires de files d'attente de passerelle configurés de manière identique pour améliorer la disponibilité d'une passerelle. Supposons que vous ayez défini des files d'attente d'alias de cluster sur les passerelles pour une file d'attente locale définie dans le cluster. Si la file d'attente locale devient indisponible, vous souhaitez que le message soit conservé sur l'une des passerelles en attendant que la file d'attente redevienne disponible. Pour suspendre la file d'attente sur une passerelle, vous devez définir la file d'attente locale avec un rang supérieur à celui des files d'attente d'alias de cluster sur la passerelle.

Si vous définissez la file d'attente locale avec le même rang que les alias de file d'attente et que la file d'attente locale n'est pas disponible, le message circule entre les passerelles. Lors de la recherche de la file d'attente locale indisponible, le premier gestionnaire de files d'attente de passerelle achemine le message vers l'autre passerelle. L'autre passerelle tente à nouveau de distribuer le message à la file d'attente locale cible. Si la file d'attente locale n'est toujours pas disponible, elle achemine le message vers la première passerelle. Le message continue d'être déplacé entre les passerelles jusqu'à ce que la file d'attente locale cible soit à nouveau disponible. En attribuant à la file d'attente locale un rang supérieur, même si la file d'attente n'est pas disponible, le message n'est pas redirigé vers une destination de rang inférieur.

IBM MQ obtient le rang des files d'attente avant de vérifier le statut du canal. L'obtention du rang avant la vérification de l'état du canal signifie que même les files d'attente non accessibles sont disponibles pour la sélection. Il permet aux messages d'être acheminés via le réseau même si la destination finale n'est pas disponible.

Si vous avez utilisé l'attribut de priorité que IBM MQ sélectionne entre les destinations disponibles. Si un canal n'est pas disponible pour la destination dont le rang est le plus élevé, le message est conservé dans la file d'attente de transmission. Il est libéré lorsque le canal devient disponible. Le message n'est pas envoyé à la destination disponible suivante dans l'ordre de classement.

CLWLUSEQ

L'attribut de file d'attente **CLWLUSEQ** indique si une instance locale d'une file d'attente est privilégiée en tant que destination par rapport à d'autres instances d'un cluster.

L'attribut de file d'attente **CLWLUSEQ** est valide uniquement pour les files d'attente locales. Elle s'applique uniquement si le message est inséré par une application ou un canal qui n'est pas un canal de cluster.

LOCAL

La file d'attente locale est la seule cible de MQPUT, à condition que la file d'attente locale soit activée. Le comportement de MQPUT dépend de la [gestion de la charge de travail du cluster](#).

QMGR

Le comportement est celui spécifié par l'attribut de gestionnaire de files d'attente **CLWLUSEQ**.

Tout

MQPUT traite la file d'attente locale comme toute autre instance de la file d'attente dans le cluster pour la distribution de la charge de travail.

Concepts associés

[Algorithme de gestion de la charge de travail du cluster](#)

L'algorithme de gestion de la charge de travail utilise des attributs d'équilibrage de charge et de nombreuses règles pour sélectionner la destination finale des messages placés dans les files d'attente de cluster.

Référence associée

[Attributs de canal d'équilibrage de charge de cluster](#)

Liste alphabétique des attributs de canal utilisés dans l'équilibrage de charge de cluster.

[Équilibrage de charge de cluster-Attributs de gestionnaire de files d'attente](#)

Liste alphabétique des attributs de gestionnaire de files d'attente utilisés dans l'équilibrage de charge de cluster.

«Attributs de cluster disponibles dans les commandes de définition de file d'attente», à la page 136
[Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente](#).

Equilibrage de charge de cluster-Attributs de gestionnaire de files d'attente

Liste alphabétique des attributs de gestionnaire de files d'attente utilisés dans l'équilibrage de charge de cluster.

CLWLMRUC

L'attribut de gestionnaire de files d'attente **CLWLMRUC** définit le nombre de canaux les plus récemment choisis. L'algorithme de gestion de charge de travail de cluster utilise **CLWLMRUC** pour limiter le nombre de canaux de cluster sortants actifs. La valeur doit être comprise entre 1 et 999 999 999.

La valeur par défaut initiale est 999 999 999.

CLWLUSEQ

L'attribut de gestionnaire de files d'attente **CLWLUSEQ** indique si une instance locale d'une file d'attente est privilégiée en tant que destination par rapport à d'autres instances de la file d'attente dans un cluster. L'attribut s'applique si l'attribut de file d'attente **CLWLUSEQ** est défini sur QMGR.

L'attribut de file d'attente **CLWLUSEQ** est valide uniquement pour les files d'attente locales. Elle s'applique uniquement si le message est inséré par une application ou un canal qui n'est pas un canal de cluster.

LOCAL

La file d'attente locale est la seule cible de MQPUT. LOCAL est la valeur par défaut.

Tout

MQPUT traite la file d'attente locale comme toute autre instance de la file d'attente dans le cluster pour la distribution de la charge de travail.

Concepts associés

Algorithme de gestion de la charge de travail du cluster

L'algorithme de gestion de la charge de travail utilise des attributs d'équilibrage de charge et de nombreuses règles pour sélectionner la destination finale des messages placés dans les files d'attente de cluster.

Référence associée

Attributs de canal d'équilibrage de charge de cluster

Liste alphabétique des attributs de canal utilisés dans l'équilibrage de charge de cluster.

Equilibrage de charge de cluster-Attributs de file d'attente

Liste alphabétique des attributs de file d'attente utilisés dans l'équilibrage de charge de cluster.

«Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente», à la page 139

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

Algorithme de gestion de la charge de travail du cluster

L'algorithme de gestion de la charge de travail utilise des attributs d'équilibrage de charge et de nombreuses règles pour sélectionner la destination finale des messages placés dans les files d'attente de cluster.

L'algorithme de gestion de la charge de travail est appliqué chaque fois qu'un choix de destination est requis:

- Il est utilisé au moment où une file d'attente de cluster est ouverte, à l'aide de l'option MQOO_BIND_ON_OPEN .
- Il est utilisé chaque fois qu'un message est inséré dans une file d'attente de cluster lorsqu'il est ouvert avec MQOO_BIND_NOT_FIXED.
- Il est utilisé chaque fois qu'un nouveau groupe de messages est démarré lorsque MQOO_BIND_ON_GROUP est utilisé pour ouvrir une file d'attente de cluster.

- Pour le roulage via un hôte de rubrique, il est utilisé chaque fois qu'un message est publié dans une rubrique en cluster. Si le gestionnaire de files d'attente local n'est pas un hôte pour cette rubrique, l'algorithme est utilisé pour choisir un gestionnaire de files d'attente hôte pour acheminer le message.

La section suivante décrit l'algorithme de gestion de charge de travail utilisé pour déterminer la destination finale des messages placés dans les files d'attente de cluster. Ces règles sont influencées par les paramètres appliqués aux attributs suivants pour les files d'attente, les gestionnaires de files d'attente et les canaux:

Files d'attente	Gestionnaires de files d'attente	Canaux
<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</u> • <u>NETPRTY</u>

Initialement, le gestionnaire de files d'attente génère une liste de destinations possibles à partir de deux procédures:

- Mise en correspondance de la cible `ObjectName` et `ObjectQmgrName` avec les définitions d'alias de gestionnaire de files d'attente qui sont partagées dans les mêmes clusters que le gestionnaire de files d'attente.
- Recherche de routes uniques (c'est-à-dire de canaux) vers un gestionnaire de files d'attente qui héberge une file d'attente nommée `ObjectName` et qui se trouve dans l'un des clusters dont le gestionnaire de files d'attente est membre.

L'algorithme suit les règles suivantes pour éliminer les destinations de la liste des destinations possibles.

1. Les instances distantes des files d'attente ou des rubriques ou les canaux `CLUSRCVR` distants qui ne partagent pas de cluster avec le gestionnaire de files d'attente local sont éliminés.
2. Si un nom de file d'attente ou de rubrique est indiqué, les canaux `CLUSRCVR` éloignés qui ne se trouvent pas dans le même cluster que la file d'attente ou la rubrique sont éliminés.

Remarque : Toutes les files d'attente, rubriques et canaux restants à ce stade sont mises à la disposition de l'exit de charge de travail de cluster, s'il est configuré.

3. Tous les canaux vers les gestionnaires de files d'attente ou les alias de gestionnaire de files d'attente dont la valeur `CLWLRANK` est inférieure au rang maximal de tous les canaux ou alias de gestionnaire de files d'attente restants sont éliminés.
4. Toutes les files d'attente (qui ne sont pas des alias de gestionnaire de files d'attente) dont la valeur `CLWLRANK` est inférieure au rang maximal de toutes les files d'attente restantes sont éliminées.
5. Si plusieurs instances d'une file d'attente, d'une rubrique ou d'un alias de gestionnaire de files d'attente sont conservées et si une publication est activée, toutes celles qui sont désactivées sont éliminées.

Remarque : Si seules les instances d'insertion désactivées sont conservées, seules les opérations d'interrogation aboutissent, toutes les autres opérations échouent avec `MQRC_CLUSTER_PUT_DISABLED`.

6. Lors du choix d'une file d'attente, si l'ensemble de résultats des files d'attente contient l'instance locale de la file d'attente, l'instance locale est généralement utilisée. L'instance locale de la file d'attente est utilisée si l'une des conditions suivantes est remplie:

- L'attribut `use-queue` de la file d'attente, `CLWLUSEQ`, est défini sur `LOCAL`.
- Les deux affirmations suivantes sont vraies:

¹ Cet attribut s'applique uniquement lors du choix d'une file d'attente en cluster et non lors du choix d'une rubrique.

- L'attribut use-queue de la file d'attente, CLWLUSEQ, est défini sur QMGR.
 - L'attribut use-queue du gestionnaire de files d'attente, CLWLUSEQ, est défini sur LOCAL.
 - Le message est reçu via un canal de cluster plutôt qu'en étant inséré par une application locale.
 - Pour les files d'attente définies en local qui sont définies avec CLWLUSEQ (ANY), ou qui héritent de ce même paramètre du gestionnaire de files d'attente, les points suivants sont vrais, dans l'ensemble plus large de conditions qui s'appliquent:
 - La file d'attente locale est choisie en fonction du statut des canaux CLUSRCVR définis localement dans le même cluster que la file d'attente. Ce statut est comparé au statut des canaux CLUSSDR qui envoient le message aux files d'attente définies à distance du même nom.

Par exemple, il existe un CLUSRCVR dans le même cluster que la file d'attente. Ce CLUSRCVR est à l'état STOPPING, alors que les autres files d'attente du même nom dans le cluster sont à l'état RUNNING ou INACTIVE. Dans ce cas, les canaux distants seront choisis et les canaux CLUSSDR locaux ne seront pas utilisés.
 - La file d'attente locale est choisie en fonction du nombre des canaux CLUSRCVR, en toute comparaison avec les canaux CLUSSDR du même statut, qui achemine le message vers des files d'attente définies à distance du même nom.

Par exemple, il existe quatre canaux CLUSRCVR dans le même cluster que la file d'attente et un canal CLUSSDR. Tous les canaux ont le même statut INACTIVE ou RUNNING. Par conséquent, il y a cinq canaux à choisir et deux instances de la file d'attente. Les quatre cinquièmes (80%) des messages sont placés dans la file d'attente locale.
7. Si plusieurs gestionnaires de files d'attente sont conservés, s'ils ne sont pas interrompus, tous ceux qui sont suspendus sont éliminés.
 8. Si plusieurs instances distantes d'une file d'attente ou d'une rubrique sont conservées, tous les canaux inactifs ou en cours d'exécution sont inclus. Les constantes d'état sont répertoriées:
 - MQCHS_XX_ENCODE_CASE_ONE inactif
 - MQCHS_EN COURS D'EXECUTION
 9. Si aucune instance distante d'une file d'attente ou d'une rubrique n'est conservée, tous les canaux en cours de liaison, d'initialisation, de démarrage ou d'arrêt sont inclus. Les constantes d'état sont répertoriées:
 - LIAISON-mqchs_binding
 - MQCHS_INITIALISATION
 - MQCHS_DEMARRAGE
 - MQCHS_ARRÊT
 10. Si aucune instance éloignée d'une file d'attente ou d'une rubrique n'est conservée, tous les canaux qui font l'objet d'une nouvelle tentative sont inclus. La constante d'état est répertoriée:
 - MQCHS_RELANCE
 11. Si aucune instance éloignée d'une file d'attente ou d'une rubrique n'est conservée, tous les canaux à l'état de demande, de mise en pause ou d'arrêt sont inclus. Les constantes d'état sont répertoriées:
 - MQCHS_DEMANDE
 - MQCHS_PAUSED
 - MQCHS_ARRETE
 - MQCHS_COMMUTATION
 12. Si plusieurs instances distantes d'une file d'attente ou d'une rubrique d'un gestionnaire de files d'attente sont conservées, les canaux ayant la valeur NETPRTY la plus élevée pour chaque gestionnaire de files d'attente sont sélectionnés.
 13. Tous les canaux restants et les alias de gestionnaire de files d'attente autres que les canaux et les alias ayant la priorité la plus élevée, CLWLPRTY, sont éliminés. S'il reste des alias de gestionnaire de files d'attente, les canaux vers le gestionnaire de files d'attente sont conservés.

14. Si une file d'attente est choisie:

- Toutes les files d'attente autres que celles dont la priorité est la plus élevée, CLWLPRTY, sont éliminées et les canaux sont conservés.

15. Les canaux restants sont ensuite réduits au maximum au nombre maximal autorisé de canaux les plus récemment utilisés, CLWLMRUC, en éliminant les canaux ayant les valeurs les plus faibles de MQWDR.DestSeqNumber.

Remarque : Les messages de contrôle de cluster internes sont envoyés à l'aide du même algorithme de charge de travail de cluster, le cas échéant.

Une fois que la liste des destinations valides a été calculée, la charge de travail des messages est équilibrée entre elles, à l'aide de la logique suivante:

- Lorsque plusieurs instances distantes d'une destination sont conservées et que CLWLWGHT est défini sur la valeur par défaut 50 pour tous les canaux de cette destination, le canal le moins récemment utilisé est choisi. Cela correspond approximativement à un style circulaire d'équilibrage de la charge de travail lorsque plusieurs instances distantes existent.
- Lorsque plusieurs instances distantes d'une destination restent et qu'un ou plusieurs des canaux de ces files d'attente ont CLWLWGHT défini sur un paramètre non par défaut (même s'ils ont tous une valeur non par défaut correspondante), le routage devient dépendant des pondérations relatives de chaque canal et du nombre total de fois que chaque canal a été choisi lors de l'envoi de messages.
- Lorsque vous observez la distribution des messages pour une seule file d'attente en cluster avec plusieurs instances, cela peut donner lieu à une distribution non équilibrée dans un sous-ensemble d'instances de file d'attente. En effet, c'est l'utilisation historique de chaque canal émetteur de cluster de ce gestionnaire de files d'attente qui est équilibrée, et pas seulement le trafic de messages de cette file d'attente. Si ce comportement n'est pas souhaitable, effectuez l'une des opérations suivantes:
 - Définissez CLWLWGHT sur 50 sur tous les canaux récepteurs de cluster si une distribution uniforme est requise.
 - Ou, si certaines instances de file d'attente doivent être pondérées différemment des autres, définissez ces files d'attente dans un cluster dédié, avec des canaux récepteurs de cluster dédiés définis. Cette action isole l'équilibrage de charge de ces files d'attente des autres dans le cluster.
- Les données d'historique utilisées pour équilibrer les canaux sont réinitialisées si des attributs de charge de travail de cluster des canaux récepteurs de cluster disponibles sont modifiés ou si le statut d'un canal récepteur de cluster devient disponible. La modification des attributs de charge de travail des canaux émetteurs de cluster définis manuellement ne réinitialise pas les données d'historique.
- Lorsque vous prenez en compte la logique d'exit de charge de travail de cluster, le canal choisi est celui avec le MQWDR MQWDR.DestSeqFactor. Chaque fois qu'un canal est choisi, cette valeur est augmentée d'environ 1000/CLWLWGHT. S'il existe plusieurs canaux avec la valeur la plus faible, l'un des canaux avec le MQWDR MQWDR.DestSeqNumber est choisie.

La distribution des messages utilisateur n'est pas toujours exacte car l'administration et la maintenance du cluster entraînent la circulation des messages entre les canaux. Il en résulte une distribution inégale des messages utilisateur qui peut prendre un certain temps pour se stabiliser. En raison de la combinaison des messages d'administration et d'utilisateur, ne vous reposez pas sur la distribution exacte des messages lors de l'équilibrage de la charge de travail.

Référence associée

Attributs de canal d'équilibrage de charge de cluster

Liste alphabétique des attributs de canal utilisés dans l'équilibrage de charge de cluster.

Équilibrage de charge de cluster-Attributs de file d'attente

Liste alphabétique des attributs de file d'attente utilisés dans l'équilibrage de charge de cluster.

Équilibrage de charge de cluster-Attributs de gestionnaire de files d'attente

Liste alphabétique des attributs de gestionnaire de files d'attente utilisés dans l'équilibrage de charge de cluster.

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

[Equilibrage de charge dans les clusters](#)

Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Il utilise l'algorithme de gestion de la charge de travail du cluster et un certain nombre d'attributs spécifiques à la charge de travail du cluster pour déterminer le meilleur gestionnaire de files d'attente à utiliser.

Related tasks

[Checking that async commands for distributed networks have finished](#)

Related reference

[Attributs de cluster disponibles dans les commandes de définition de canal](#)

Attributs de cluster pouvant être spécifiés dans les commandes de définition de canal.

[Attributs de cluster disponibles dans les commandes de définition de file d'attente](#)

Attributs de cluster pouvant être spécifiés dans les commandes de définition de file d'attente.

[Attributs de cluster disponibles dans les commandes de définition de gestionnaire de files d'attente](#)

Attributs de cluster pouvant être spécifiés dans les commandes de définition de gestionnaire de files d'attente.

DISPLAY CLUSQMGR

La commande DISPLAY CLUSQMGR permet d'afficher des informations de cluster sur les gestionnaires de files d'attente d'un cluster.

Actualiser le cluster

Emettez la commande REFRESH CLUSTER à partir d'un gestionnaire de files d'attente pour supprimer toutes les informations détenues localement sur un cluster. Il est peu probable que vous ayez à utiliser cette commande, sauf dans des circonstances exceptionnelles.

RESET CLUSTER: Suppression forcée d'un gestionnaire de files d'attente d'un cluster

La commande **RESET CLUSTER** permet de supprimer de force un gestionnaire de files d'attente d'un cluster dans des circonstances exceptionnelles.

SUSPEND QMGR, RESUME QMGR et clusters

Utilisez la commande SUSPEND QMGR et RESUME QMGR pour réduire temporairement l'activité de cluster entrante dans ce gestionnaire de files d'attente, par exemple, avant d'effectuer la maintenance sur ce gestionnaire de files d'attente, puis de le rétablir.

Programmes de canal

Cette section examine les différents types de programmes de canal (MCM) disponibles sur les canaux.

Les noms des MCM sont indiqués dans les tableaux suivants.

Tableau 61. Programmes de canal pour les systèmes AIX, Linux, and Windows

Nom du programme	Sens de la connexion	Communication
amqrmppa		Tous
runmqlsr	Entrant	Tous
amqcrs6a	Entrant	LU 6.2
amqcrsta	Entrant	TCP
runmqchl	Sortant	Tous
runmqchi	Sortant	Tous

runmqlsr (Exécution du programme d'écoute IBM MQ), runmqchl (Exécution du canal IBM MQ) et runmqchi (Exécution de l'initiateur de canal IBM MQ) sont des commandes de contrôle que vous pouvez entrer sur la ligne de commande.

amqcrsta est appelé pour les canaux TCP sur les systèmes AIX and Linux utilisant inetd, où aucun programme d'écoute n'est démarré.

amqcrs6a est appelé en tant que programme de transaction lors de l'utilisation de LU6.2

IBM i Travaux d'intercommunication sous IBM i

Les travaux suivants sont associés à Intercommunication sur IBM i. Les noms sont contenus dans le tableau suivant.

Tableau 62. Noms et descriptions de travail

Nom du travail	Description
AMQCLMAA	Prog. écoute sans unité d'exécution
amqcrsta	Travail canal rép. sans unité d'exéc
AMQRMPPA	Travail du groupe de canaux
runmqchi	Initialisateur de canal
runmqchl	Travail du canal
runmqlsr	Prog. écoute des unités d'exécution

IBM i Etats des canaux sous IBM i

Les états des canaux sont affichés dans le panneau Gestion des canaux

Tableau 63. Etats des canaux sous IBM i

Nom de l'état	Explication
DEMARRAGE EN COURS	Le canal est prêt à commencer la négociation avec l'agent MCA cible
BINDING	Etablissement d'une session et échange initial de données
Demande en cours	Canal demandeur initiant une connexion
EN COURS D'EXECUTION	Transfert ou prêt à être transféré
PAUSE	En attente de l'intervalle entre les nouvelles tentatives de message
EN COURS D'ARRET	Etablissement d'une nouvelle tentative ou d'un arrêt
Nouvel essai	En attente jusqu'à la prochaine tentative de relance

Tableau 63. Etats des canaux sous IBM i (suite)

Nom de l'état	Explication
ARRETE	Canal arrêté en raison d'une erreur ou d'une commande de canal d'extrémité émise
INACTIF	Le traitement du canal s'est arrêté normalement ou le canal n'a jamais démarré
*Aucun	Aucun état (pour les canaux de connexion serveur uniquement)

ALW Exemple: planification d'un canal de transmission de messages sur AIX, Linux, and Windows

Ces informations fournissent un exemple détaillé de connexion de deux gestionnaires de files d'attente entre eux afin que des messages puissent être envoyés entre eux.

Pourquoi et quand exécuter cette tâche

Dans tous les exemples, les commandes MQSC sont affichées telles qu'elles apparaissent dans un fichier de commandes et telles qu'elles sont saisies sur la ligne de commande. Les deux méthodes sont identiques, mais pour émettre une commande sur la ligne de commande, vous devez d'abord entrer `runmqsc`, pour le gestionnaire de files d'attente par défaut, ou `runmqsc qmname`, où `qmname` est le nom du gestionnaire de files d'attente requis. Entrez ensuite un nombre quelconque de commandes, comme indiqué dans les exemples.

Une autre méthode consiste à créer un fichier contenant ces commandes. Toute erreur dans les commandes est alors facile à corriger. Si vous avez appelé votre fichier `mqsc.in`, pour l'exécuter sur le gestionnaire de files d'attente `QMNAME`, utilisez:

```
runmqsc QMNAME < mqsc.in > mqsc.out
```

Vous pouvez vérifier les commandes dans votre fichier avant de l'exécuter à l'aide de la commande suivante:

```
runmqsc -v QMNAME < mqsc.in > mqsc.out
```

Pour des raisons de portabilité, vous devez limiter la longueur de ligne de vos commandes à 72 caractères. Utilisez un caractère de concaténation pour continuer sur plusieurs lignes. Sous Windows, utilisez Ctrl-z pour terminer l'entrée sur la ligne de commande. Sur les systèmes AIX and Linux, utilisez Ctrl-d. Vous pouvez également utiliser la commande **end**.

La Figure 7, à la page 158 illustre l'exemple de scénario.

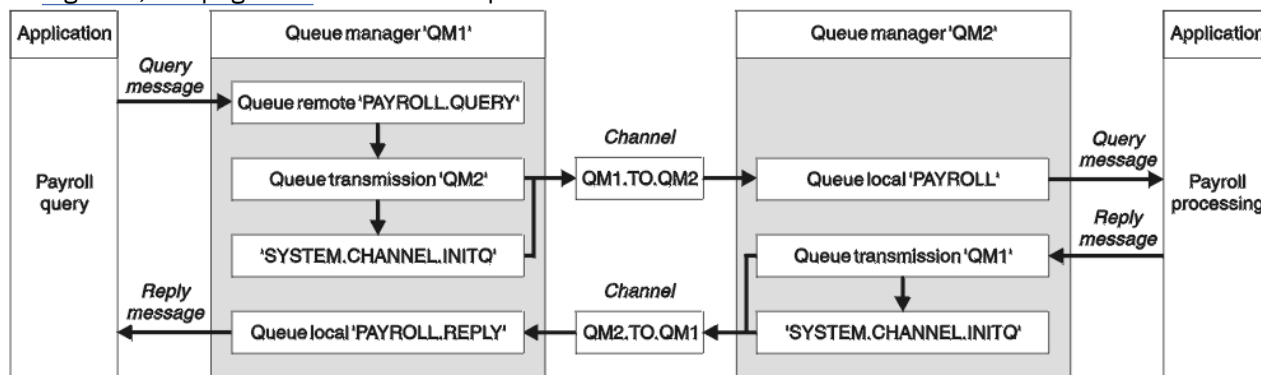


Figure 7. Exemple de canal de transmission de messages pour les systèmes AIX, Linux, and Windows

Cet exemple implique une application de requête de paie connectée au gestionnaire de files d'attente QM1 qui envoie des messages de requête de paie à une application de traitement de paie s'exécutant sur le gestionnaire de files d'attente QM2. L'application de requête de paie a besoin des réponses à ses requêtes renvoyées à QM1. Les messages de requête de paie sont envoyés de QM1 à QM2 sur un canal émetteur-récepteur appelé QM1.TO.QM2 et les messages de réponse sont renvoyés de QM2 à QM1 sur un autre canal émetteur-récepteur appelé QM2.TO.QM1. Ces deux canaux sont déclenchés pour démarrer dès qu'ils ont un message à envoyer à l'autre gestionnaire de files d'attente.

L'application de requête de paie insère un message de requête dans la file d'attente éloignée "PAYROLL.QUERY" défini sur QM1. Cette définition de file d'attente éloignée est convertie en file d'attente locale "PAYROLL" sur QM2. En outre, l'application de requête de paie indique que la réponse à la requête est envoyée à la file d'attente locale "PAYROLL.REPLY" sur QM1. L'application de traitement de paie extrait les messages de la file d'attente locale "PAYROLL" sur QM2 et envoie les réponses partout où elles sont requises ; dans ce cas, la file d'attente locale "PAYROLL.REPLY" sur QM1.

Dans les exemples de définitions pour TCP/IP, QM1 a l'adresse d'hôte 192.0.2.0 et est en mode écoute sur le port 1411, et QM2 a l'adresse d'hôte 192.0.2.1 et est en mode écoute sur le port 1412. L'exemple suppose qu'ils sont déjà définis sur votre système et qu'ils peuvent être utilisés.

Les définitions d'objet qui doivent être créées sur QM1 sont les suivantes:

- Définition de file d'attente éloignée, PAYROLL.QUERY
- Définition de file d'attente de transmission, QM2 (par défaut = nom du gestionnaire de files d'attente éloignées)
- Définition de canal émetteur, QM1.TO.QM2 QM2
- Définition de canal récepteur, QM2.TO.QM1
- Définition de file d'attente de réponse, PAYROLL.REPLY

Les définitions d'objet qui doivent être créées sur QM2 sont les suivantes:

- Définition de file d'attente locale, PAYROLL
- Définition de file d'attente de transmission, QM1 (par défaut = nom du gestionnaire de files d'attente éloignées)
- Définition de canal émetteur, QM2.TO.QM1
- Définition de canal récepteur, QM1.TO.QM2 QM2

Les détails de connexion sont fournis dans l'attribut CONNAME des définitions de canal émetteur.

Vous pouvez voir un diagramme de la disposition dans [Figure 7](#), à la page 158.

Procédure

Voir :

- [«Configuration de l'exemple de canal de transmission de messages pour AIX, Linux, and Windows»](#), à la page 159 pour plus de détails sur la configuration des canaux de transmission de messages
- [«Exécution et développement de l'exemple pour AIX, Linux, and Windows»](#), à la page 161 pour obtenir des suggestions sur la façon dont vous pouvez utiliser d'autres produits, par exemple CICS, et sur la façon dont vous pouvez connecter davantage d'applications et d'exits utilisateur.

ALW Configuration de l'exemple de canal de transmission de messages pour AIX, Linux, and Windows

Ces définitions d'objet permettent aux applications connectées au gestionnaire de files d'attente QM1 d'envoyer des messages de demande à une file d'attente appelée PAYROLL sur QM2 et de recevoir des réponses sur une file d'attente appelée PAYROLL.REPLY sur QM1, et permet aux applications connectées au gestionnaire de files d'attente QM2 d'extraire les messages de demande d'une file d'attente locale appelée PAYROLL, et d'insérer les réponses à ces messages de demande dans une file d'attente appelée PAYROLL.REPLY sur le gestionnaire de files d'attente QM1.

Pourquoi et quand exécuter cette tâche

Toutes les définitions d'objet ont été fournies avec les attributs DESCR et REPLACE. Les autres attributs fournis sont le minimum requis pour que les exemples fonctionnent. Les attributs qui ne sont pas fournis prennent les valeurs par défaut pour les gestionnaires de files d'attente QM1 et QM2.

Il n'est pas nécessaire de fournir une définition de file d'attente éloignée pour permettre le renvoi des réponses à QM1. Le descripteur de message du message extrait de la file d'attente locale PAYROLL contient à la fois le nom de la file d'attente de réponse et le nom du gestionnaire de files d'attente de réponse. Par conséquent, tant que QM2 peut résoudre le nom du gestionnaire de files d'attente de réponse en celui d'une file d'attente de transmission sur le gestionnaire de files d'attente QM2, le message de réponse peut être envoyé. Dans cet exemple, le nom du gestionnaire de files d'attente de réponse est QM1. Par conséquent, le gestionnaire de files d'attente QM2 requiert une file d'attente de transmission du même nom.

Procédure

- Exécutez les commandes suivantes sur le gestionnaire de files d'attente QM1:
 - a) Configurez la définition de file d'attente éloignée:

```
DEFINE QREMOTE(PAYROLL.QUERY) DESCR('Remote queue for QM2') REPLACE +  
PUT(ENABLED) XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Remarque : La définition de file d'attente éloignée n'est pas une file d'attente physique, mais un moyen de diriger les messages vers la file d'attente de transmission, QM2, afin qu'ils puissent être envoyés au gestionnaire de files d'attente QM2.

- b) Configurez la définition de file d'attente de transmission:

```
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)
```

Lorsque le premier message est inséré dans cette file d'attente de transmission, un message de déclenchement est envoyé à la file d'attente d'initialisation, SYSTEM.CHANNEL.INITQ. L'initiateur de canal extrait le message de la file d'attente d'initialisation et démarre le canal identifié dans le processus nommé.

- c) Configurez la définition de canal émetteur:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) TRPTYPE(TCP) +  
REPLACE DESCR('Sender channel to QM2') XMITQ(QM2) +  
CONNAME('192.0.2.1(1412)')
```

- d) Configurez la définition de canal récepteur:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(RCVR) TRPTYPE(TCP) +  
REPLACE DESCR('Receiver channel from QM2')
```

- e) Configurez la définition de la file d'attente de réponse:

```
DEFINE QLOCAL(PAYROLL.REPLY) REPLACE PUT(ENABLED) GET(ENABLED) +  
DESCR('Reply queue for replies to query messages sent to QM2')
```

La file d'attente de réponse est définie comme PUT (ENABLED). Ainsi, les messages de réponse peuvent être placés dans la file d'attente. Si les réponses ne peuvent pas être placées dans la file d'attente de réponses, elles sont envoyées à la file d'attente de rebut sur QM1 ou, si cette file d'attente n'est pas disponible, restent dans la file d'attente de transmission QM1 sur le gestionnaire de files d'attente QM2. La file d'attente a été définie en tant que GET (ENABLED) pour permettre l'extraction des messages de réponse.

- Exécutez les commandes suivantes sur le gestionnaire de files d'attente QM2:
 - a) Configurez la définition de file d'attente locale:

```
DEFINE QLOCAL(PAYROLL) REPLACE PUT(ENABLED) GET(ENABLED) +
DESCR('Local queue for QM1 payroll details')
```

Cette file d'attente est définie comme PUT (ENABLED) et GET (ENABLED) pour la même raison que la définition de file d'attente de réponse sur le gestionnaire de files d'attente QM1.

- b) Configurez la définition de file d'attente de transmission:

```
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)
```

Lorsque le premier message est inséré dans cette file d'attente de transmission, un message de déclenchement est envoyé à la file d'attente d'initialisation, SYSTEM.CHANNEL.INITQ. L'initiateur de canal extrait le message de la file d'attente d'initialisation et démarre le canal identifié dans le processus nommé.

- c) Configurez la définition de canal émetteur:

```
DEFINE CHANNEL(QM2.TO.QM1) CHLTYPE(SDR) TRPTYPE(TCP) +
REPLACE DESCR('Sender channel to QM1') XMITQ(QM1) +
CONNAME('192.0.2.0(1411)')
```

- d) Configurez la définition de canal récepteur:

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(RCVR) TRPTYPE(TCP) +
REPLACE DESCR('Receiver channel from QM1')
```

ALW Exécution et développement de l'exemple pour AIX, Linux, and Windows

Informations sur le démarrage de l'initiateur de canal et du programme d'écoute et suggestions pour développer ce scénario.

Pourquoi et quand exécuter cette tâche

Une fois ces définitions créées, vous devez:

- Démarrez l'initiateur de canal sur chaque gestionnaire de files d'attente.
- Démarrez le programme d'écoute pour chaque gestionnaire de files d'attente.

Vous pouvez également développer l'exemple.

Procédure

1. Démarrez l'initiateur de canal et le programme d'écoute.

Voir [Configuration de la communication pour Windows](#) et [Configuration de la communication sur les systèmes AIX and Linux](#).

2. Vous pouvez développer cet exemple en:

- Utilisation des communications LU 6.2 pour l'interconnexion avec les systèmes CICS et le traitement des transactions.
- Ajout de définitions de file d'attente, de processus et de canal supplémentaires pour permettre à d'autres applications d'envoyer des messages entre les deux gestionnaires de files d'attente.

- Ajout de programmes d'exit utilisateur sur les canaux pour permettre le chiffrement des liens, la vérification de la sécurité ou un traitement supplémentaire des messages.
- Utilisation d'alias de gestionnaire de files d'attente et d'alias de file d'attente de réponse pour en savoir plus sur leur utilisation dans l'organisation de votre réseau de gestionnaires de files d'attente.

IBM i Exemple: planification d'un canal de transmission de messages sur IBM i

Exemple détaillé de connexion de deux gestionnaires de files d'attente IBM i ensemble afin que des messages puissent être envoyés entre eux. Cet exemple implique une application de requête de paie connectée au gestionnaire de files d'attente QM1 qui envoie des messages de requête de paie à une application de traitement de paie s'exécutant sur le gestionnaire de files d'attente QM2. L'application de requête de paie a besoin des réponses à ses requêtes renvoyées à QM1.

Pourquoi et quand exécuter cette tâche

L'exemple illustre les préparations nécessaires pour permettre à une application utilisant le gestionnaire de files d'attente QM1 d'insérer des messages dans une file d'attente du gestionnaire de files d'attente QM2. Une application exécutée sur QM2 peut extraire ces messages et envoyer des réponses à une file d'attente de réponses sur QM1.

L'exemple illustre l'utilisation des connexions TCP/IP. L'exemple suppose que les canaux doivent être déclenchés pour démarrer lorsque le premier message arrive dans la file d'attente de transmission qu'ils traitent.

Cet exemple utilise SYSTEM.CHANNEL.INITQ comme file d'attente d'initialisation. Cette file d'attente est déjà définie par IBM MQ. Vous pouvez utiliser une autre file d'attente d'initialisation, mais vous devez la définir vous-même, démarrer une nouvelle instance de l'initiateur de canal à l'aide de la commande STRMQMCHLI et la fournir avec le nom de votre file d'attente d'initialisation. Pour plus d'informations sur le déclenchement de canaux, voir [Déclenchement de canaux](#).

Remarque : MQ Adv. CD Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

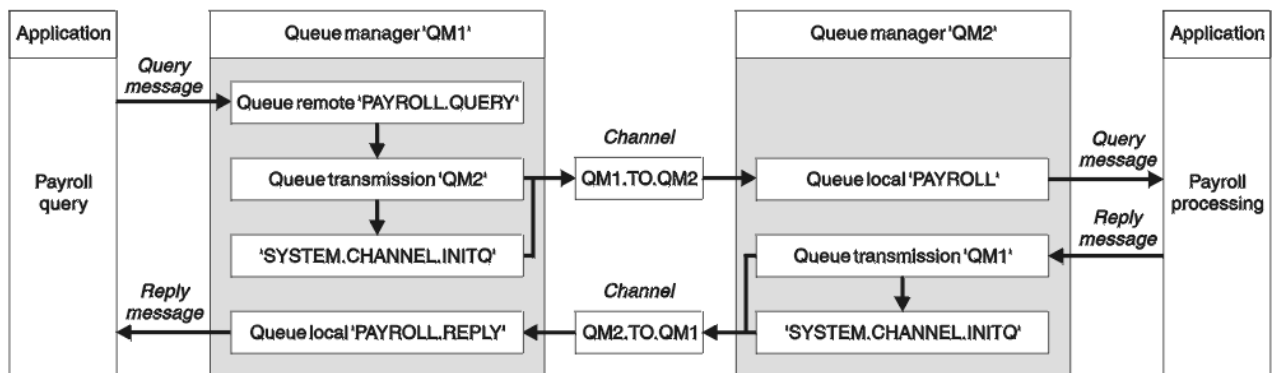


Figure 8. Exemple de canal de transmission de messages pour IBM MQ for IBM i

Les messages de requête de paie sont envoyés de QM1 à QM2 sur un canal émetteur-récepteur appelé QM1.TO.QM2 et les messages de réponse sont renvoyés de QM2 à QM1 sur un autre canal émetteur-récepteur appelé QM2.TO.QM1. Ces deux canaux sont déclenchés pour démarrer dès qu'ils ont un message à envoyer à l'autre gestionnaire de files d'attente.

L'application de requête de paie insère un message de requête dans la file d'attente éloignée "PAYROLL.QUERY" défini sur QM1. Cette définition de file d'attente éloignée est convertie en file d'attente locale "PAYROLL" sur QM2. En outre, l'application de requête de paie indique que la réponse à la requête

est envoyée à la file d'attente locale "PAYROLL.REPLY" sur QM1. L'application de traitement de paie extrait les messages de la file d'attente locale "PAYROLL" sur QM2 et envoie les réponses partout où elles sont requises ; dans ce cas, la file d'attente locale "PAYROLL.REPLY" sur QM1.

Les deux gestionnaires de files d'attente sont supposés être en cours d'exécution sur IBM i. Dans les exemples de définition, QM1 possède l'adresse d'hôte 192.0.2.0 et est à l'écoute sur le port 1411. QM2 possède l'adresse d'hôte 192.0.2.1 et est en mode écoute sur le port 1412. L'exemple suppose que ces gestionnaires de files d'attente sont déjà définis sur votre système IBM i et qu'ils peuvent être utilisés.

Les définitions d'objet qui doivent être créées sur QM1 sont les suivantes:

- Définition de file d'attente éloignée, PAYROLL.QUERY
- Définition de file d'attente de transmission, QM2 (par défaut = nom du gestionnaire de files d'attente éloignées)
- Définition de canal émetteur, QM1.TO.QM2 QM2
- Définition de canal récepteur, QM2.TO.QM1
- Définition de file d'attente de réponse, PAYROLL.REPLY

Les définitions d'objet qui doivent être créées sur QM2 sont les suivantes:

- Définition de file d'attente locale, PAYROLL
- Définition de file d'attente de transmission, QM1 (par défaut = nom du gestionnaire de files d'attente éloignées)
- Définition de canal émetteur, QM2.TO.QM1
- Définition de canal récepteur, QM1.TO.QM2 QM2

Les détails de connexion sont fournis dans l'attribut CONNAME des définitions de canal émetteur.

Vous pouvez voir un diagramme de la disposition dans [Figure 8](#), à la page 162.

Procédure

Voir :

- «[Configuration de l'agent MCA sous IBM i](#)», à la page 163 pour plus de détails sur la configuration des canaux de transmission de messages
- «[Exécution et développement de l'exemple pour IBM i](#)», à la page 166 pour obtenir des suggestions sur la façon de connecter davantage d'applications et d'exits utilisateur.

IBM i Configuration de l'agent MCA sous IBM i

Les définitions d'objet suivantes permettent aux applications connectées au gestionnaire de files d'attente QM1 d'envoyer des messages de demande à une file d'attente appelée PAYROLL sur QM2 afin de recevoir des réponses sur une file d'attente appelée PAYROLL.REPLY sur QM1, permet aux applications connectées au gestionnaire de files d'attente QM2 d'extraire les messages de demande d'une file d'attente locale appelée PAYROLL et d'insérer les réponses à ces messages de demande dans une file d'attente appelée PAYROLL.REPLY sur le gestionnaire de files d'attente QM1.

Pourquoi et quand exécuter cette tâche

Toutes les définitions d'objet ont été fournies avec les attributs TEXT. Les autres attributs fournis sont le minimum requis pour que l'exemple fonctionne. Les attributs qui ne sont pas fournis prennent les valeurs par défaut pour les gestionnaires de files d'attente QM1 et QM2.

Il n'est pas nécessaire de fournir une définition de file d'attente éloignée pour permettre le renvoi des réponses à QM1. Le descripteur de message du message extrait de la file d'attente locale PAYROLL contient à la fois le nom de la file d'attente de réponse et le nom du gestionnaire de files d'attente de réponse. Par conséquent, tant que QM2 peut résoudre le nom du gestionnaire de files d'attente de réponse en celui d'une file d'attente de transmission sur le gestionnaire de files d'attente QM2, le

message de réponse peut être envoyé. Dans cet exemple, le nom du gestionnaire de files d'attente de réponse est QM1 . Par conséquent, le gestionnaire de files d'attente QM2 requiert une file d'attente de transmission du même nom.

Procédure

- Exécutez les commandes suivantes sur le gestionnaire de files d'attente QM1:
 - a) Configurez la définition de file d'attente éloignée à l'aide de la commande CRTMQMQ avec les attributs suivants:

QNAME	'PAYROLL.QUERY'
QTYPE	*RMT
TEXTE	'File d'attente éloignée pour QM2'
PUTENBL	*YES
NOMMQT	'QM2'(par défaut = nom du gestionnaire de files d'attente éloignées)
RMTQNAME	'PAIE'
RMTMQMNAME	'QM2'

Remarque : La définition de file d'attente éloignée n'est pas une file d'attente physique, mais un moyen de diriger les messages vers la file d'attente de transmission, QM2, afin qu'ils puissent être envoyés au gestionnaire de files d'attente QM2.

- b) Configurez la définition de file d'attente de transmission à l'aide de la commande CRTMQMQ avec les attributs suivants:

QNAME	QM2
QTYPE	*LCL
TEXTE	'File d'attente de transmission vers QM2'
SYNTAXE	*TMQ
PUTENBL	*YES
Commande GETENBL	*YES
TRGENBL	*YES
TRGTYPE	*FIRST
NOM D'INITQ	SYSTEM.CHANNEL.INITQ
TRIGDATA	QM1.TO.QM2

Lorsque le premier message est inséré dans cette file d'attente de transmission, un message de déclenchement est envoyé à la file d'attente d'initialisation, SYSTEM.CHANNEL.INITQ. L'initiateur de canal extrait le message de la file d'attente d'initialisation et démarre le canal identifié dans le processus nommé.

- c) Configurez la définition de canal émetteur à l'aide de la commande CRTMQCHL avec les attributs suivants:

NOM DU CLHA	QM1.TO.QM2
CHLTYPE	*SDR
TRPTYPE	*TCP
TEXTE	'Canal émetteur vers QM2'
NOMMQT	QM2

CONNAME '192.0.2.1(1412)'

- d) Configurez la définition de canal récepteur à l'aide de la commande CRTMQCHL avec les attributs suivants:

NOM DU CLHA QM2.TO.QM1
CHLTYPE *RCVR
TRPTYPE *TCP
TEXTE 'Canal récepteur de QM2'

- e) Configurez la définition de file d'attente de réponse à l'aide de la commande CRTMQMQ avec les attributs suivants:

QNAME PAYROLL.REPLY
QTYPE *LCL
TEXTE 'File d'attente de réponses aux messages de requête envoyés à QM2'
PUTENBL *YES
Commande GETENBL *YES

La file d'attente de réponse est définie comme PUT (ENABLED). Cette définition garantit que les messages de réponse peuvent être placés dans la file d'attente. Si les réponses ne peuvent pas être placées dans la file d'attente de réponses, elles sont envoyées à la file d'attente de rebut sur QM1 ou, si cette file d'attente n'est pas disponible, restent dans la file d'attente de transmission QM1 sur le gestionnaire de files d'attente QM2. La file d'attente a été définie en tant que GET (ENABLED) pour permettre l'extraction des messages de réponse.

- Exécutez les commandes suivantes sur le gestionnaire de files d'attente QM2:

- a) Configurez la définition de file d'attente locale à l'aide de la commande CRTMQMQ avec les attributs suivants:

QNAME payroll
QTYPE *LCL
TEXTE 'File d'attente locale pour les détails de paie QM1 '
PUTENBL *YES
Commande GETENBL *YES

Cette file d'attente est définie comme PUT (ENABLED) et GET (ENABLED) pour la même raison que la définition de file d'attente de réponse sur le gestionnaire de files d'attente QM1.

- b) Configurez la définition de file d'attente de transmission à l'aide de la commande CRTMQMQ avec les attributs suivants:

QNAME QM1
QTYPE *LCL
TEXTE 'File d'attente de transmission vers QM1'
SYNTAXE *TMQ
PUTENBL *YES
Commande GETENBL *YES
TRGENBL *YES
TRGTYPE *FIRST
NOM D'INITQ SYSTEM.CHANNEL.INITQ

TRIGDATA QM2.TO.QM1

Lorsque le premier message est inséré dans cette file d'attente de transmission, un message de déclenchement est envoyé à la file d'attente d'initialisation, SYSTEM.CHANNEL.INITQ. L'initiateur de canal extrait le message de la file d'attente d'initialisation et démarre le canal identifié dans les données de déclenchement.

- c) Configurez la définition de canal émetteur à l'aide de la commande CRTMQMCHL avec les attributs suivants:

NOM DU CLHA	QM2.TO.QM1
CHLTYPE	*SDR
TRPTYPE	*TCP
TEXTE	'Canal émetteur vers QM1'
NOMMQT	QM1
CONNAME	'192.0.2.0(1411)'

- d) Configurez la définition de canal récepteur à l'aide de la commande CRTMQMCHL avec les attributs suivants:

NOM DU CLHA	QM1.TO.QM2
CHLTYPE	*RCVR
TRPTYPE	*TCP
TEXTE	'Canal récepteur de QM1'

IBM i

Exécution et développement de l'exemple pour IBM i

Informations sur le démarrage de l'initiateur de canal et du programme d'écoute et suggestions pour développer ce scénario.

Pourquoi et quand exécuter cette tâche

Une fois ces définitions créées, vous devez:

- Démarrez l'initiateur de canal sur chaque gestionnaire de files d'attente.
- Démarrez le programme d'écoute pour chaque gestionnaire de files d'attente.

Les applications peuvent alors s'envoyer des messages les unes aux autres. Les canaux sont déclenchés pour démarrer par le premier message arrivant sur chaque file d'attente de transmission, de sorte que vous n'avez pas besoin d'émettre la commande STRMQMCHL.

Vous pouvez également développer l'exemple.

Procédure

1. Démarrez l'initiateur de canal et le programme d'écoute.

Pour plus d'informations sur le démarrage d'un initiateur de canal et d'un programme d'écoute, voir [Surveillance et contrôle des canaux sur IBM i](#).

2. Vous pouvez développer cet exemple en:

- Ajout de définitions de file d'attente et de canal supplémentaires pour permettre à d'autres applications d'envoyer des messages entre les deux gestionnaires de files d'attente.
- Ajout de programmes d'exit utilisateur sur les canaux pour permettre le chiffrement des liens, la vérification de la sécurité ou un traitement supplémentaire des messages.

- Utilisation d'alias de gestionnaire de files d'attente et d'alias de file d'attente de réponse pour mieux comprendre comment ces objets peuvent être utilisés dans l'organisation de votre réseau de gestionnaires de files d'attente.

► **z/OS** Pour une version de cet exemple qui utilise des commandes MQSC, voir «[Exemple: planning a message channel on z/OS](#)», à la page 167.

► **z/OS** 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: ► **MQ Adv.** ► **CD** Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

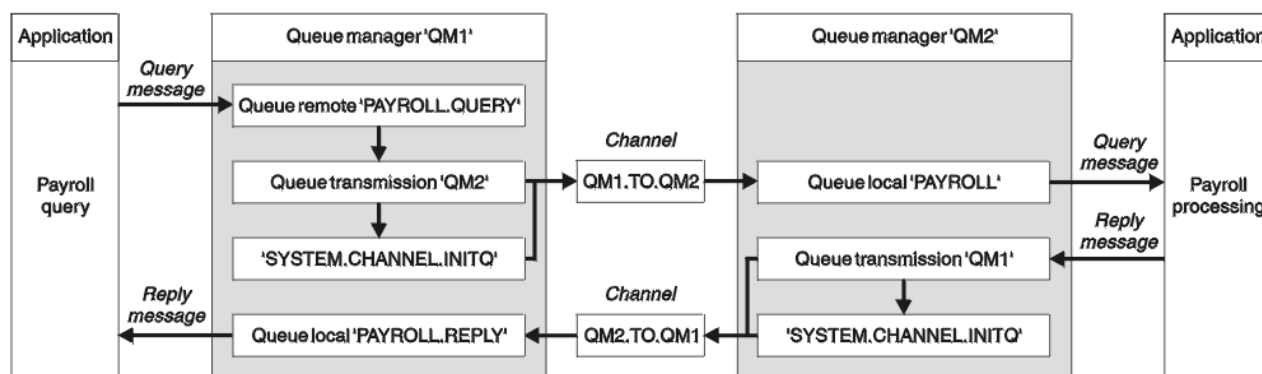


Figure 9. The first example for IBM MQ for z/OS

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 48.

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 167.

Procedure

See:

- [“Setting up the message channel agent on z/OS”](#) on page 168 for details on setting up the message channels
- [“Running and expanding the example for z/OS”](#) on page 170 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) +
CONNAME('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) +
CONNAME('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 167](#) 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.

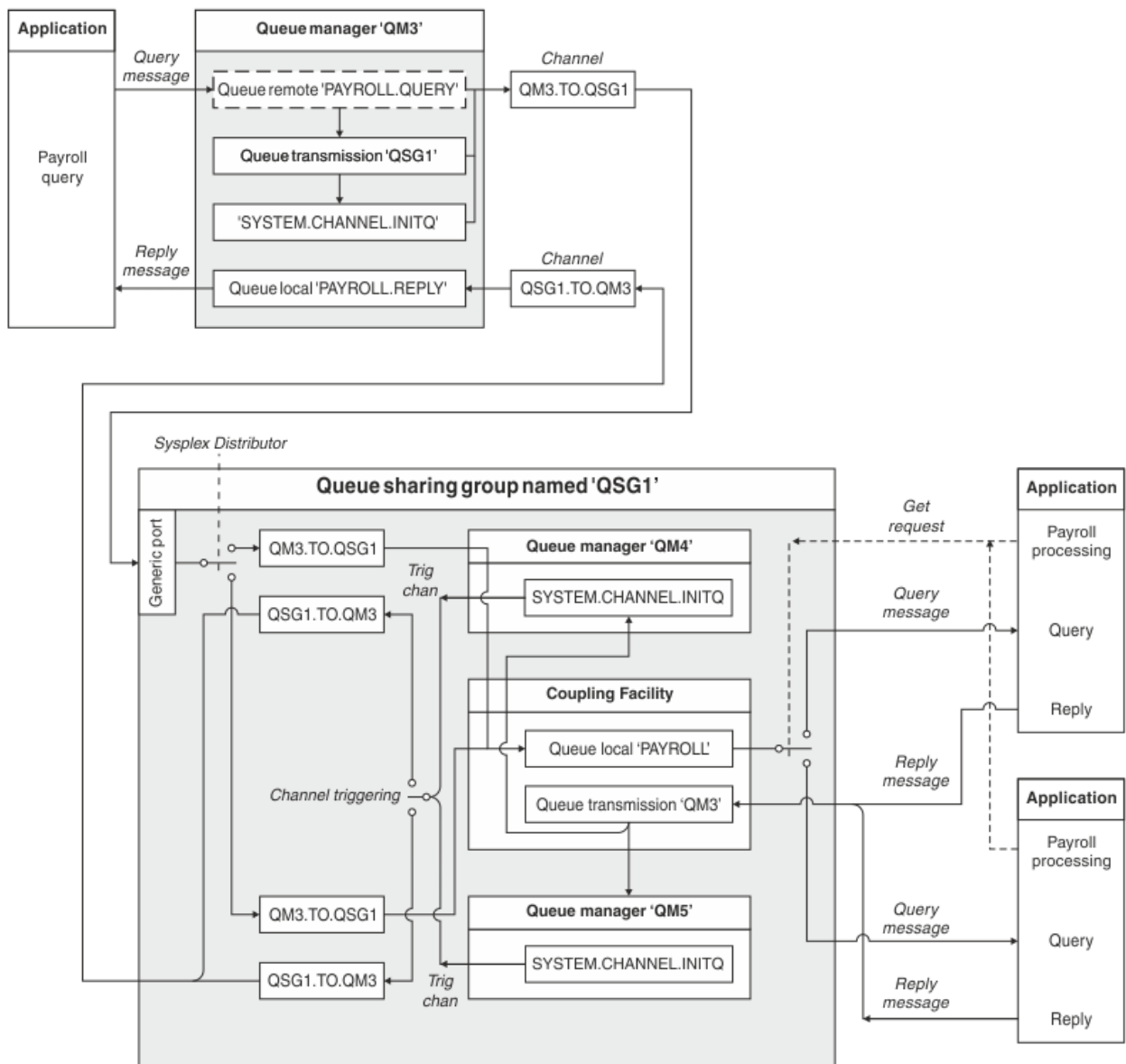


Figure 10. Message channel planning example for IBM MQ for z/OS using queue sharing groups

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 53.

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 173 for details on setting up the definitions.

- “Running the queue sharing group example for z/OS” on page 174 for details on starting the channel initiators and listeners for each queue manager.

z/OS 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 167.

```
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.

Utilisation d'un alias pour faire référence à une bibliothèque MQ

Vous pouvez définir un alias pour faire référence à une bibliothèque MQ dans votre JCL, au lieu d'utiliser directement le nom de la bibliothèque MQ. Ensuite, si le nom de la bibliothèque MQ change, il vous suffit de supprimer et de redéfinir l'alias.

Exemple

L'exemple suivant définit un alias MQM.SCSQANLE pour faire référence à la bibliothèque MQ MQM.V600.SCSQANLE:

```
//STEP1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DELETE (MQM.SCSQANLE)
DEFINE ALIAS (NAME(MQM.SCSQANLE) RELATE(MQM.V600.SCSQANLE))
/*
```

Ensuite, pour faire référence à MQM.V600.SCSQANLE dans votre JCL, utilisez l'alias MQM.SCSQANLE.

Remarque : Les noms de bibliothèque et d'alias doivent se trouver dans le même catalogue. Utilisez donc le même qualificatif de haut niveau pour les deux ; dans cet exemple, le qualificatif de haut niveau est MQM.

Managed File Transfer Référence de configuration

Informations de référence pour vous aider à configurer Managed File Transfer.

Utilisation des variables d'environnement dans les propriétés MFT

Il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction de l'environnement en cours (par exemple, l'utilisateur exécutant une commande).

Les propriétés suivantes acceptent les emplacements de fichier ou de répertoire et peuvent donc contenir des variables d'environnement:

- 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

Exemple pour Windows

Windows Dans cet exemple sur un système Windows , un utilisateur fteuser utilisant une variable d'environnement USERPROFILE:

```
wmqfte.database.credentials.file=%USERPROFILE%\logger\mqmftcredentials.xml
```

Résout le chemin d'accès au fichier suivant:

```
C:\Users\fteuser\logger\mqmftcredentials.xml
```

Exemple pour AIX and Linux

Linux **AIX** Dans cet exemple sur un système UNIX , un utilisateur fteuser utilisant une variable d'environnement HOME:

```
transferRoot=$HOME/fte/
```

Résout le chemin d'accès au fichier suivant:

```
/home/fteuser/fte/
```

Référence associée

«Le fichier MFT coordination.properties», à la page 205

Le fichier `coordination.properties` indique les détails de connexion au gestionnaire de file d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier `coordination.properties` commun sur une unité partagée.

«Le fichier MFT `command.properties`», à la page 210

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

«Le fichier MFT `agent.properties`», à la page 181

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

Propriétés SSL/TLS pour MFT

«Le fichier MFT `logger.properties`», à la page 214

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Propriétés de l'agent MFT pour les exits utilisateur

Format du fichier de propriétés du pont de protocole

Format de fichier de définition de processus Connect:Direct

Format de fichier de propriétés du noeud Connect:Direct

Le fichier MFT `installation.properties`

Le fichier `installation.properties` indique le nom de votre ensemble d'options de configuration par défaut. Cette entrée renvoie Managed File Transfer vers un ensemble structuré de répertoires et de fichier de propriétés qui contiennent la configuration à utiliser. Généralement, le nom d'un ensemble d'options de configuration est le nom du gestionnaire de files d'attente de coordination associé.

Ce fichier est créé par le programme d'installation et peut être modifié à l'aide de la commande **`fteChangeDefaultConfigurationOptions`**.

Le fichier `installation.properties` se trouve dans votre répertoire `MQ_DATA_PATH`.

Par exemple, sous Windows, l'emplacement de fichier par défaut est

`MQ_DATA_PATH\mqft\installations\installation_name` et sur les systèmes AIX and Linux,

l'emplacement de fichier par défaut est `/var/mqm/mqft/installations/installation_name`.

Pour l'Redistributable Managed File Transfer Agent, le chemin de données est défini lorsque vous exécutez la commande **`fteCreateEnvironment`**. Si vous exécutez la commande et indiquez l'emplacement de votre choix à l'aide du paramètre **`-d`**, le chemin de données est défini pour cet emplacement. Si vous ne spécifiez pas l'emplacement à l'aide de la commande **`fteCreateEnvironment`**, un répertoire `mftdata` est créé sous le répertoire racine de l'extraction de l'Redistributable Managed File Transfer Agent. Le fichier `installation.properties` pour Redistributable Managed File Transfer Agent se trouve dans le répertoire `MQ_DATA_PATH\mqft\installations\MFTZipInstall`.

Le fichier `installation.properties` contient les valeurs suivantes :




Tableau 64. Propriétés de base

Nom de la propriété	Description	Valeur par défaut
commandMessagePriority	<p>Définit la priorité des messages internes et des messages de commande des commandes fteStopAgent, fteCancelTransfer et ftePingAgent.</p> <p>Si, par exemple, vous soumettez un grand nombre de demandes de transfert pour transférer plein de petits fichiers par succession rapide, les nouvelles demandes de transfert peuvent être placées en file d'attente, dans la file d'attente des commandes de l'agent source. Les messages externes et internes possédant la priorité de message IBM MQ par défaut, les messages internes sont bloqués par les nouvelles demandes de transfert. Cela peut entraîner un dépassement du délai de négociation des transferts et un passage des transferts en récupération.</p> <p>Vous pouvez également utiliser la propriété commandMessagePriority pour définir la priorité des messages d'accusé de réception internes et d'accusé de réception attendu.</p> <p>Pour définir les priorités des messages Managed File Transfer internes au dessus des nouvelles demandes de transfert, définissez cette propriété sur une valeur comprise entre 1 (la plus faible) et 9 (la plus élevée).</p> <p>La valeur par défaut de la propriété commandMessagePriority est 8. Cela signifie que si l'attribut IBM MQ DEFPRTY (priorité par défaut) dans une file d'attente de commandes d'agent est inférieur ou égal à 7, les messages de négociation interne sont classés par priorité avant les nouvelles demandes de transfert. Si la valeur de l'attribut DEFPRTY est de 8 ou 9, pour conserver l'efficacité de la propriété commandMessagePriority, vous devez modifier DEFPRTY ou la propriété commandMessagePriority.</p>	La valeur par défaut est 8.

Tableau 64. Propriétés de base (suite)

Nom de la propriété	Description	Valeur par défaut
commonCredentialsKeyFile	<p>Nom de chemin qualifié complet du fichier contenant la clé de données d'identification utilisée lors du chiffrement des données d'identification. Le nom le plus courant du fichier de données d'identification MFT est <code>MQMFTCredentials.xml</code>.</p> <p>Pour plus d'informations sur l'utilisation de la propriété <code>commonCredentialsKeyFile</code>, voir Déchiffrement des données d'identification.</p>	Chemin d'accès complet du fichier de clés
defaultProperties	Nom de l'ensemble par défaut d'options de configuration. Cette valeur correspond au nom d'un répertoire qui se trouve dans le répertoire de configuration, qui contient les répertoires et fichiers de propriétés qui spécifient les informations de configuration.	Aucune valeur par défaut
enableFunctionalFixPack	<p>Niveau de fonction du groupe de correctifs à activer. Par défaut, les nouvelles fonctions incluses dans un groupe de correctifs ne sont pas activées. Définissez cette propriété sur un identificateur de version pour activer les nouvelles fonctions disponibles avec cette version.</p> <p>Vous pouvez spécifier l'identificateur de version avec ou sans point (.). Par exemple, pour utiliser la fonction disponible avec IBM MQ 8.0.0 Fix Pack 2, définissez cette propriété sur <code>8002</code> ou <code>8.0.0.2</code>.</p>	Aucune valeur par défaut

Tableau 64. Propriétés de base (suite)

Nom de la propriété	Description	Valeur par défaut
messagePublicationFormat	<p>Permet de spécifier le format de publication des messages utilisé par les agents MFT pour leurs messages XML de statut. Cette propriété accepte les valeurs suivantes :</p> <p>messagePublicationFormat=mixed Les messages sont publiés sans FORMAT MQMD (MQFMT_NONE), excepté pour les messages publiés sous l'arborescence de rubriques / LOG, qui sont publiés au format MQFMT_STRING de MQMD.</p> <p>messagePublicationFormat=MQFMT_NONE Les messages sont publiés sans FORMAT MQMD.</p> <p>messagePublicationFormat=MQFMT_STRING Les messages sont publiés dans un format de chaîne.</p>	messagePublicationFormat=mixed
<p> z/OS:</p>		
productId	<p>Type de produit sur lequel l'utilisation de MFT doit être enregistrée :</p> <ul style="list-style-type: none"> • Produit Managed File Transfer autonome. (MFT correspond à productID). • Composant d'un produit IBM MQ Advanced. (ADVANCED correspond à productID). •  Composant d'un produit IBM MQ Advanced for z/OS Value Unit Edition. (ADVANCEDVUE correspond à productID). <p>Pour plus d'informations sur l'enregistrement de l'utilisation du produit, voir Génération de rapports sur les informations produit .</p> <p> Multi Cette propriété est ignorée sous Multiplateformes.</p>	MFT

Le texte suivant représente un exemple de contenu d'un fichier `installation.properties`.

```
defaultProperties=ERIS
```

ERIS représente le nom d'un répertoire qui se trouve dans le même répertoire que le fichier `installation.properties`. Le répertoire ERIS contient des répertoires et des fichiers de propriétés qui décrivent un ensemble d'options de configuration.

Concepts associés

[Options de configuration de MFT sur Multiplatforms](#)

Référence associée

[fteChangeDefaultConfigurationOptions](#)

Le fichier MFT agent.properties

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

Le fichier `agent.properties` est créé par le programme d'installation ou par la commande **`fteCreateAgent`**, **`fteCreateBridgeAgent`** ou **`fteCreateCDAgent`**. Vous pouvez utiliser l'une de ces commandes avec l'option **`-f`** pour modifier les propriétés de base du gestionnaire de files d'attente de l'agent, ainsi que les propriétés d'agent avancées associées au type d'agent que vous créez. Pour modifier ou ajouter des propriétés d'agent avancées, vous devez éditer le fichier dans un éditeur de texte.

Multi Sur Multiplatforms, le fichier `agent.properties` d'un agent se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name`.

z/OS Sous z/OS, l'emplacement du fichier `agent.properties` est `$BFG_CONFIG variable/mqft/config/coordination_qmgr_name/agents/agent_name`.

Si vous modifiez le fichier `agent.properties`, vous devez redémarrer l'agent pour appliquer vos modifications.

Vous pouvez utiliser des variables d'environnement dans certaines propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela vous permet d'utiliser les emplacements de fichiers ou de répertoires lors de l'exécution de composants du produit en fonction des modifications d'environnement (par exemple, en fonction de l'utilisateur qui exécute le processus). Pour plus d'informations, voir [«Utilisation des variables d'environnement dans les propriétés MFT»](#), à la page 175.

Windows

Remarque : Sous Windows, deux propriétés:

- `windowsService`
- `windowsServiceVersion`

sont ajoutés dans le fichier `agent.properties` par les commandes MFT utilisées pour configurer un agent à exécuter en tant que service Windows.

Vous ne devez pas ajouter les propriétés, ni les modifier manuellement, car cela empêchera l'agent de fonctionner correctement.

Pour plus d'informations sur la configuration d'un agent pour qu'il démarre en tant que service Windows, voir [Démarriage d'un agent MFT en tant que service Windows](#).

Propriétés de l'agent de base

Chaque fichier MFT `agent.properties` contient les propriétés d'agent de base suivantes:


Tableau 65. Propriétés de l'agent de base

Nom de la propriété	Description	Valeur par défaut
agentName	Nom de l'agent. Le nom de l'agent doit être conforme aux conventions de dénomination des objets IBM MQ. Pour plus d'informations, voir «Conventions de dénomination des objets MFT», à la page 242.	Aucune valeur par défaut
agentDesc	Description de l'agent (si vous choisissez de créer une description).	Aucune valeur par défaut
agentQMgr	Nom du gestionnaire de files d'attente de l'agent.	Aucune valeur par défaut
agentQMgrHost	Nom d'hôte ou adresse IP du gestionnaire de files d'attente de l'agent.	Aucune valeur par défaut
agentQMgrPort	Numéro de port utilisé pour établir les connexions client avec le gestionnaire de files d'attente de l'agent.	1414
agentQMgrChannel	Nom de canal SVRCONN utilisé pour se connecter au gestionnaire de files d'attente de l'agent.	SYSTEM.DEF.SVRCONN
agentType	Type d'agent : <ul style="list-style-type: none"> • Agent standard autre qu'un agent de pont (STANDARD) • Agent de pont de protocole (BRIDGE) • Agent de pont Connect:Direct (CD_BRIDGE) • Agent imbriqué tel qu'il est utilisé par IBM Integration Bus (EMBEDDED) • Agent imbriqué à SFG (Sterling File Gateway) 	STANDARD

Si vous ne spécifiez pas de valeur pour la propriété agentQMgrHost, le mode liaisons est utilisé par défaut.

Si vous spécifiez une valeur pour la propriété agentQMgrHost, mais que vous n'en spécifiez pas pour les propriétés agentQMgrPort et agentQMgrChannel, le numéro de port 1414 et le canal SYSTEM.DEF.SVRCONN sont utilisés par défaut.

Propriétés d'agent avancées

Managed File Transfer fournit également davantage de propriétés d'agent avancées qui vous aident à configurer les agents. Si vous souhaitez utiliser l'une des propriétés ci-après, éditez manuellement le fichier `agent.properties` pour ajouter les propriétés avancées requises. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappées à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Pour plus d'informations sur la mise en échappement des caractères dans les fichiers de propriétés Java, voir la documentation Oracle [Javadoc pour la classe Properties](#).

- [Propriétés de taille d'agent](#)
- [Propriétés de page de codes](#)
- [Propriétés de commande](#)
- [Propriétés de connexion](#)
- [Propriétés de pontConnect:Direct](#)
- [Propriétés d'agent de fichier à message et de message à fichier](#)
- [Propriétés d'agent générales](#)
- [Propriétés de haute disponibilité](#)
- [Propriétés d'entrée/de sortie](#)
- [Propriétés du journal de transfert](#)
- [Propriétés de prise en charge multicanal](#)
- [Propriétés multiinstance](#)


- Propriétés de contrôleur de processus
- Propriétés de pont de protocole
- Propriétés de journal d'agent de pont de protocole
- Propriétés de file d'attente
- Propriétés de surveillance des ressources
- Propriétés du répertoire racine
- Propriété Scheduler
- Propriétés de sécurité
- Propriétés SSL/TLS
- Propriétés de délai d'expiration
- Propriétés de délai d'expiration pour la reprise des transferts
- Propriétés de trace et de consignation
- Propriétés de limite de transfert
- Propriétés de routine d'exit utilisateur
- Propriétés de compression du client IBM MQ
-  Propriétés spécifiques à z/OS
- Autres propriétés

Tableau 66. Propriétés d'agent avancées: taille de l'agent

Nom de la propriété	Description	Valeur par défaut
agentCheckpointInterval	<p>Intervalle en cadres de données complets au cours duquel un point de contrôle est effectué à des fins de récupération. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>En cas de problème entraînant le passage du transfert en récupération, le transfert ne peut être restauré que sur une limite de point de contrôle. Par conséquent, plus cette valeur est élevée (avec des valeurs agentChunkSize, agentWindowSize et agentFrameSize élevées), plus l'agent a besoin de temps pour récupérer les transferts. Pour les réseaux Managed File Transfer fiables où les transferts ne passent que rarement à l'état de récupération, il peut être judicieux d'augmenter cette valeur pour augmenter les performances globales.</p>	1
agentChunkSize	<p>Taille de chaque bloc de transfert pour le transport des données de fichier. Indique donc la taille maximale des messages IBM MQ transférés entre les agents source et de destination. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>Cette valeur est négociée entre les agents source et de destination, la valeur la plus élevée des deux étant utilisée. Si vous souhaitez modifier la valeur de cette propriété, remplacez-la au niveau de l'agent source et de l'agent de destination.</p> <p>agentChunkSize est une valeur entière. Par exemple: agentChunkSize = 10240 définit la taille de bloc sur 10 Ko.</p>	262144-byte (équivalent à 256 ko)
agentFrameSize	<p>Nombre de fenêtres pour le cadre de transfert. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>Pour les réseaux dont le temps d'attente est élevé, l'augmentation de cette valeur peut améliorer les performances globales car elle permet à l'agent d'avoir plus de blocs de message actifs simultanément.</p> <p>La valeur de cette propriété, multipliée par celle d'agentWindowSize, multipliée par celle d'agentChunkSize, indique la limite supérieure de la consommation de mémoire de l'agent pour chaque transfert. Par exemple, blocs de 262 144 octets x 10 x 5 = 12,5 Mo par transfert.</p> <p>Remarque : si la taille des fichiers transférés au cours d'un même transfert est inférieure à 12,5 Mo, l'augmentation de la valeur de cette propriété n'a aucune incidence sur les performances du transfert.</p>	5

Tableau 66. Propriétés d'agent avancées: taille de l'agent (suite)

Nom de la propriété	Description	Valeur par défaut
agentWindowSize	<p>Nombre de blocs de chaque fenêtre. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur.</p> <p>Pour les réseaux dont le temps d'attente est élevé, l'augmentation de cette valeur peut améliorer les performances globales. Cela est dû au fait que l'agent a plus de blocs de messages actifs simultanément et réduit la fréquence à laquelle les messages d'accusé de réception sont renvoyés à l'agent source.</p> <p>La valeur de cette propriété, multipliée par la taille agentFrame, multipliée par la taille agentChunk, indique la limite supérieure de la consommation de mémoire de l'agent pour chaque transfert et indique la limite supérieure des données de message IBM MQ dans la file d'attente de données de l'agent de destination. Par exemple, blocs de 262 144 octets x 10 x 5 = limite supérieure de 12,5 Mo par transfert.</p> <p>Remarque : si la taille des fichiers transférés au cours d'un même transfert est inférieure à 12,5 Mo, l'augmentation de la valeur de cette propriété n'a aucune incidence sur les performances du transfert.</p>	10

Tableau 67. Propriétés d'agent avancées: page de codes

Nom de la propriété	Description	Valeur par défaut
agentCcsid	Page de codes avec laquelle l'agent se connecte à son gestionnaire de files d'attente d'agent. Si vous spécifiez une valeur pour agentCcsid, vous devez également en spécifier une pour agentCcsidName. Pour plus d'informations sur la manière d'afficher les pages de codes connues pour la machine virtuelle Java, reportez-vous au paramètre -hsc de la commande fteCreateBridgeAgent .	1208
agentCcsidName	Représentation Java d'agentCcsid. Si vous spécifiez une valeur pour agentCcsidName, vous devez également en spécifier une pour agentCcsid.	UTF8

Tableau 68. Propriétés d'agent avancées: Commande

Nom de la propriété	Description	Valeur par défaut
maxCommandHandlerThreads	Contrôle le nombre d'unités d'exécution disponibles pour l'analyse initiale et le traitement des messages de commande de transfert. Lorsqu'elles sont actives, les unités d'exécution requièrent une connexion au gestionnaire de files d'attente, mais elles la libèrent lorsqu'elles sont inactives.	5
maxCommandOutput	Nombre maximal d'octets stockés pour la sortie de commande. Cette propriété s'applique aux commandes spécifiées pour un appel géré et aux commandes preSource, postSource, preDestination et postDestination d'un transfert géré. Cela permet de limiter la longueur de la sortie de commande consignée dans le journal de transfert sur la rubrique SYSTEM.FTE.	10240
maxCommandRetries	Nombre maximal de nouvelles tentatives d'une commande autorisées par l'agent. Cette propriété s'applique aux commandes spécifiées pour un appel géré et aux commandes preSource, postSource, preDestination et postDestination d'un transfert géré.	9
maxCommandWait	Délai d'attente maximal, en secondes, entre les nouvelles tentatives autorisées par l'agent. Cette propriété s'applique aux commandes spécifiées pour un appel géré et aux commandes preSource, postSource, preDestination et postDestination d'un transfert géré.	60
immediateShutdownTimeout	<p>Pour un arrêt immédiat d'un agent, vous pouvez utiliser cette propriété pour indiquer la durée maximale en secondes pendant laquelle un agent attend la fin de ses transferts avant de forcer un arrêt.</p> <p>Remarque : Ne spécifiez pas une valeur inférieure à la valeur par défaut de 10 secondes pour cette propriété. Un arrêt immédiat d'un agent requiert suffisamment de temps pour arrêter les éventuels processus externes. Si la valeur de cette propriété est trop faible, des processus peuvent rester actifs.</p> <p>Si la valeur 0 est spécifiée pour cette propriété, l'agent attend que tous les transferts en attente soient arrêtés. Si une valeur non valide est spécifiée pour cette propriété, la valeur par défaut est utilisée.</p>	10

Nom de la propriété	Description	Valeur par défaut
javaLibraryPath	Lors de la connexion à un gestionnaire de files d'attente en mode liaisons, Managed File Transfer doit avoir accès aux bibliothèques de liaisons IBM MQ Java . Par défaut, Managed File Transfer recherche ces bibliothèques de liaisons dans l'emplacement par défaut défini par IBM MQ. Si elles se trouvent dans un autre emplacement, utilisez cette propriété pour en spécifier l'emplacement.	Aucun

Nom de la propriété	Description	Valeur par défaut
cdNode	Propriété requise si vous souhaitez utiliser le pont Connect:Direct. Nom du noeud Connect:Direct à utiliser pour transférer les messages de l'agent de pont Connect:Direct vers les noeuds Connect:Direct de destination. Ce noeud fait partie du pont Connect:Direct et non du noeud distant qui correspond à la source ou à la destination du transfert. Pour plus d'informations, voir The Connect:Direct bridge .	Aucune valeur par défaut
cdNodeHost	Nom d'hôte ou adresse IP du noeud Connect:Direct à utiliser pour transférer les fichiers de l'agent de pont Connect:Direct vers les noeuds de destination (noeud de pont Connect:Direct). Dans la plupart des cas, le noeud de pont Connect:Direct se trouve sur le même système que l'agent de pont Connect:Direct. Dans ces cas là, la valeur par défaut de cette propriété (l'adresse IP du système local) est correcte. Si votre système possède plusieurs adresses IP ou que votre noeud de pont Connect:Direct se trouve sur un autre système que votre agent de pont Connect:Direct et que leurs systèmes partagent un système de fichiers, utilisez cette propriété pour spécifier le nom d'hôte correct du noeud de pont Connect:Direct. Si vous n'avez pas défini la propriété cdNode, cette propriété est ignorée.	Nom d'hôte ou adresse IP du système local
cdNodePort	Numéro de port du noeud de pont Connect:Direct utilisé par les applications client pour communiquer avec le noeud. Dans la documentation du produit Connect:Direct, ce port est appelé port d'API. Si vous n'avez pas défini la propriété cdNode, cette propriété est ignorée.	1363
cimpDir	Emplacement permettant de stocker les fichiers temporairement sur le système où l'agent de pont Connect:Direct est exécuté avant qu'ils ne soient transférés vers le noeud Connect:Direct de destination. Cette propriété spécifie le chemin d'accès complet du répertoire dans lequel les fichiers sont stockés temporairement. Par exemple, si cdTmpDir est défini sur /tmp, les fichiers sont temporairement placés dans le répertoire /tmp. L'agent de pont Connect:Direct et le noeud de pont Connect:Direct doivent pouvoir accéder au répertoire indiqué par ce paramètre en utilisant le même nom de chemin. Tenez compte de ce point lorsque vous planifiez l'installation du pont Connect:Direct. Si possible, créez l'agent sur le système sur lequel se trouve le noeud Connect:Direct appartenant au pont Connect:Direct. Si votre agent et votre noeud se trouvent sur des systèmes distincts, le répertoire doit figurer sur un système de fichiers partagé et être accessible depuis les deux systèmes via le même nom de chemin. Pour plus d'informations sur les configurations prises en charge, voir Le pont Connect:Direct . Si vous n'avez pas défini la propriété cdNode, cette propriété est ignorée. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.	<code>value_of_java.io.tmpdir/cdbridge-agentName</code>
cdTrace	Indique si l'agent trace les données envoyées entre l'agent de pont Connect:Direct et son noeud Connect:Direct. La valeur de cette propriété peut être true ou false.	false
cdMaxConnectionRetries	Nombre maximal de tentatives de connexion Connect:Direct, pour un transfert de fichiers où aucune connexion n'a encore été établie, avant que le transfert n'échoue.	-1 (nombre infini de tentatives)
cdMaxPartialWorkConnectionRetries	Nombre maximal de tentatives de connexion Connect:Direct, pour un transfert de fichiers où une tentative de connexion précédente et le travail de transfert ont abouti, avant que le transfert n'échoue.	-1 (nombre infini de tentatives)

Tableau 70. Propriétés d'agent avancées: pont Connect:Direct (suite)


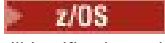




Nom de la propriété	Description	Valeur par défaut
cdMaxWaitForProcessEndStats	Délai d'attente maximal en millisecondes avant que les informations sur l'achèvement du processus Connect:Direct ne deviennent disponibles dans les statistiques du noeud Connect:Direct, une fois le processus terminé et avant que le transfert de fichier ne soit considéré comme ayant échoué. En général, ces informations sont disponibles immédiatement, mais sous certaines conditions d'échec, elles ne sont pas publiées. Dans ces conditions, le transfert de fichier échoue après le délai d'attente spécifié par cette propriété.	60000
cdAppName	Nom d'application utilisé par l'agent de pont Connect:Direct pour se connecter au noeud Connect:Direct qui fait partie du pont.	Managed File Transfer <i>version actuelle, version actuelle</i> représentant le numéro de version du produit.
cdNodeLocalPortRange	Plage de ports locaux à utiliser pour les connexion au socket entre l'agent de pont Connect:Direct et le noeud Connect:Direct qui fait partie du pont. Le format de cette valeur est une liste de valeurs ou de plages, séparées par des virgules. Par défaut, le système d'exploitation choisit les numéros de port locaux.	Aucun
cdNodeProtocol	Protocole utilisé par l'agent de pont Connect:Direct pour se connecter au noeud Connect:Direct qui fait partie du pont. Les valeurs suivantes sont valides : <ul style="list-style-type: none"> • TCPIP • SSL • TLS 	TCPIP
cdNodeKeystore	Chemin d'accès au magasin de clés utilisé pour les communications sécurisées entre l'agent de pont Connect:Direct et le noeud Connect:Direct qui fait partie du pont. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.	Aucun
cdNodeKeystoreType	Format de fichier du magasin de clés spécifié par la propriété cdNodeKeystore. Les valeurs suivantes sont valides : jks et pkcs12. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée.	jks
cdNodeKeystoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de cdNodeKeystore. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.  Pour plus de détails sur la création de fichiers de données d'identification, voir MFT et IBM MQ -Authentification de la connexion  Pour plus de détails sur la création de fichiers de données d'identification, voir Configuration de MQMFTCredentials.xml sur z/OS.	La valeur par défaut de cette propriété est:    \$HOME/MQMFTCredentials.xml  %USERPROFILE%/MQMFTCredentials.xml
cdNodeTruststore	Chemin d'accès au fichier de clés certifiées utilisé pour les communications sécurisées entre l'agent de pont Connect:Direct et le noeud Connect:Direct qui fait partie du pont. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.	Aucun
cdNodeTruststoreType	Format de fichier du fichier de clés certifiées spécifié par la propriété cdNodeTruststore. Les valeurs suivantes sont valides : jks et pkcs12. Si vous n'avez pas défini la propriété cdNodeProtocol sur SSL ou TLS, cette propriété est ignorée.	jks

Tableau 70. Propriétés d'agent avancées: pont Connect:Direct (suite)







Nom de la propriété	Description	Valeur par défaut
cdNodeTruststoreCredentialsFile	<p>Chemin d'accès au fichier qui contient les données d'identification de cdNodeTruststore.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.</p> <p> Pour plus de détails sur la création de fichiers de données d'identification, voir MFT et IBM MQ -Authentification de la connexion</p> <p> Pour plus de détails sur la création de fichiers de données d'identification, voir Configuration de MQMFTCredentials.xml sur z/OS.</p>	<p>La valeur par défaut de cette propriété est:</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows %USERPROFILE%/MQMFTCredentials.xml</p>
logCDProcess	Niveau de consignation des processus Connect:Direct enregistré dans le journal des événements d'agent, dans le fichier output0.log. Les valeurs possibles pour cette propriété sont None, Failures et All.	Aucun

Tableau 71. Propriétés avancées de l'agent: agent Fichier à message et Message à fichier

Nom de la propriété	Description	Valeur par défaut
deleteTmpFileAfterRenameFailure	Si vous définissez cette propriété sur false, les fichiers temporaires ne sont pas supprimés de la destination en cas d'échec de l'opération de changement de nom. Dans ce cas, les données transférées restent sur la destination, dans un fichier temporaire (.part). Vous pourrez renommer manuellement ce fichier ultérieurement. Par défaut, cette propriété possède la valeur true. Cette propriété s'applique à la fois aux transferts de message à fichier et de fichier à fichier.	Oui
enableQueueInputOutput	Par défaut, l'agent ne peut pas lire les données d'une file d'attente source ou écrire les données dans une file d'attente de destination dans le cadre d'un transfert. La définition de cette valeur sur true permet à l'agent d'effectuer des transferts de fichier à message et de message à fichier. La valeur de cette propriété peut être true ou false.	false
enableSystemQueueInputOutput	Indique si l'agent peut lire les données des files d'attente système d'IBM MQ ou écrire dessus. Les files d'attente système sont préfixées du qualificateur SYSTEM. Remarque : les files d'attente système sont utilisées par IBM MQ, Managed File Transfer et d'autres applications pour transmettre des informations importantes. Si vous changez cette propriété, l'agent peut accéder à ces files d'attente. Si vous activez cette propriété, utilisez les bacs à sable utilisateur pour limiter les files d'attente accessibles par l'agent.	false
enableClusterQueueInputOutput	Indique si l'agent peut lire les données des files d'attente en cluster d'IBM MQ ou écrire dessus. Remarque : Vous devez spécifier la propriété d'agent enableClusterQueueInputOutput en plus de la propriété enableQueueInputOutput.	false
maxDelimiterMatchLength	Nombre maximal de caractères pouvant correspondre avec l'expression régulière Java utilisée pour fractionner un fichier texte en plusieurs messages dans le cadre d'un transfert de fichier à message.	5
maxInputOutputMessageLength	Longueur maximale, en octets, d'un message lu à partir d'une file d'attente source ou écrit dans une file d'attente de destination par un agent. La propriété maxInputOutputMessageLength de l'agent source dans un transfert détermine le nombre d'octets pouvant être lus à partir d'un message sur la file d'attente source. La propriété maxInputOutputMessageLength de l'agent de destination dans un transfert détermine le nombre d'octets pouvant être écrits dans un message sur la file d'attente de destination. Si la longueur de ce message est supérieure à la valeur de cette propriété, le transfert échoue avec une erreur. Cette propriété n'affecte pas les files d'attente Managed File Transfer internes. Pour plus d'informations sur la modification de cette propriété, voir Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages .	1048576

Tableau 71. Propriétés avancées de l'agent: agent Fichier à message et Message à fichier (suite)

Nom de la propriété	Description	Valeur par défaut
monitorGroupRetryLimit	<p>Nombre maximal de fois qu'un moniteur déclenche à nouveau un transfert de message à fichier si le groupe de messages existe toujours dans la file d'attente. Le nombre de fois que le transfert de message à fichier est déclenché est déterminé par le nombre d'annulations MQMD du premier message dans le groupe.</p> <p>Si l'agent est redémarré, le moniteur déclenche à nouveau un transfert, même si le nombre de déclenchements de transfert dépasse la valeur de monitorGroupRetryLimit. Si, en raison de ce comportement, le nombre de fois que le transfert est déclenché dépasse la valeur de monitorGroupRetryLimit, l'agent consigne une erreur dans son journal des événements.</p> <p>Si la valeur -1 est spécifiée pour cette propriété, le moniteur déclenche à nouveau le transfert un nombre illimité de fois, jusqu'à ce que la condition de déclenchement ne soit pas satisfaite.</p>	10

Tableau 72. Propriétés d'agent avancées: Général

Nom de la propriété	Description	Valeur par défaut
agentStatusPublishRateLimit	<p>Fréquence maximale, en secondes, à laquelle l'agent republie son statut en raison d'un changement du statut du transfert de fichier.</p> <p>Si vous affectez à cette propriété une valeur trop faible, cela peut avoir un impact négatif sur les performances du réseau IBM MQ.</p>	30
agentStatusPublishRateMin	<p>Fréquence minimale, en secondes, à laquelle l'agent publie son statut. Cette valeur doit être supérieure ou égale à la valeur de la propriété agentStatusPublishRateLimit.</p>	300
enableMemoryAllocationChecking	<p>La valeur de cette propriété peut être <code>true</code> ou <code>false</code>. Il détermine si le Managed File Transfer Agent vérifie qu'il y a suffisamment de mémoire disponible pour exécuter un transfert avant qu'un transfert ne soit accepté. La vérification est effectuée à la fois sur l'agent source et sur l'agent de destination. Si la quantité de mémoire disponible est insuffisante, le transfert est rejeté.</p> <p>Lors du calcul de la quantité de mémoire requise pour un transfert, la quantité de mémoire maximale requise par le transfert est utilisée. Par conséquent, la valeur peut être supérieure à la quantité de mémoire réelle utilisée par le transfert. Pour cette raison, le nombre de transferts simultanés pouvant être exécutés peut être réduit si la propriété enableMemoryAllocationChecking est définie sur <code>true</code>. Il est recommandé de ne définir cette valeur sur <code>true</code> que si Managed File Transfer échoue avec des erreurs de mémoire insuffisante. Les transferts susceptibles de consommer des quantités importantes de mémoire sont les transferts de fichier à message et de message à fichier avec des messages de grande taille.</p>	false
enableDetailedReplyMessages	<p>La valeur de cette propriété peut être <code>true</code> ou <code>false</code>. La définition de cette propriété sur <code>true</code> permet aux réponses aux demandes de transfert gérées de contenir des informations détaillées sur les fichiers transférés. Les informations détaillées et le format sont identiques à ceux publiés dans le journal de transfert dans les messages de progression, c'est-à-dire l'élément <code><transferSet></code>. Pour plus d'informations, voir Formats de message de journal de transfert de fichier.</p> <p>Les informations de réponse détaillées ne sont incluses que si la demande de transfert gérée indique que les informations de réponse détaillées sont requises. Pour spécifier cette exigence, définissez l'attribut detailed de l'élément <code><reply></code> du message de demande XML managedTransfer envoyé à l'agent source. Pour plus d'informations, voir Format de message de demande de transfert de fichier.</p> <p>Il est possible de générer plusieurs messages de réponse par demande de transfert. Ce nombre est égal au nombre de messages de progression du journal de transfert pour le transfert, plus 1 (le premier message de réponse correspondant à une réponse d'accusé de réception simple). Les informations détaillées sont incluses dans tous les messages, exceptés les messages de réponse d'accusé de réception, mais le résultat global du transfert n'est inclus que dans le dernier message de réponse détaillé.</p>	Oui

Tableau 72. Propriétés d'agent avancées: Général (suite)

Nom de la propriété	Description	Valeur par défaut
enableUserMetadataOptions	<p>La valeur de cette propriété peut être <code>true</code> ou <code>false</code>. Il détermine si vous pouvez utiliser des clés connues pour les métadonnées définies par l'utilisateur dans les nouvelles demandes de transfert afin de fournir des options de transfert supplémentaires. Ces clés connues commencent toujours par le préfixe <code>com.ibm.wmqfte.</code>. Par conséquent, lorsque la propriété enableUserMetadataOptions est définie sur <code>true</code>, les clés qui utilisent ce préfixe ne sont pas prises en charge pour une utilisation définie par l'utilisateur. Lorsque la propriété enableUserMetadataOptions est définie sur <code>true</code>, les clés actuellement prises en charge sont les suivantes:</p> <p>com.ibm.wmqfte.insertRecordLineSeparator</p> <p>Pour les transferts de texte. Lorsque cette clé est définie sur <code>true</code>, indique que lors de la lecture de fichiers orientés enregistrement, tels que des z/OS ensembles de données, des séparateurs de ligne doivent être insérés entre les enregistrements.</p> <p>Lorsque cette clé est définie sur <code>false</code>, indique que lors de la lecture de fichiers orientés enregistrement, les séparateurs de ligne ne doivent pas être insérés entre les enregistrements.</p> <p>com.ibm.wmqfte.newRecordOnLineSeparator</p> <p>Pour les transferts de texte. Lorsque cette clé est définie sur <code>true</code>, indique que lors de l'écriture dans des fichiers orientés enregistrement, tels que des z/OS ensembles de données, les séparateurs de ligne indiquent un nouvel enregistrement et ne sont pas écrits dans le cadre des données.</p> <p>Lorsque cette clé est définie sur <code>false</code>, indique que, lors de l'écriture dans des fichiers orientés enregistrement, les séparateurs de ligne doivent être traités comme tout autre caractère (c'est-à-dire, aucun saut d'enregistrement).</p> <p>com.ibm.wmqfte.convertLineSeparators</p> <p>Pour les transferts de texte. Indique si les séquences de séparateurs de ligne CRLF et LF sont converties en séquence de séparateurs de ligne requise pour la destination. Cette conversion ne prend actuellement effet que dans les cas suivants:</p> <ul style="list-style-type: none"> • Si la clé de métadonnées définie par l'utilisateur com.ibm.wmqfte.newRecordOnLineSeparator est définie sur <code>false</code> et que le transfert est effectué dans un fichier orienté enregistrement. • Si la clé de métadonnées définie par l'utilisateur com.ibm.wmqfte.com.ibm.wmqfte.insertRecordLineSeparator est définie sur <code>false</code> et que le transfert est effectué à partir d'un fichier orienté enregistrement. <p>Voir aussi fteCreateTransfer: démarrez un nouveau transfert de fichier.</p>	false
failTransferOnFirstEchec	<p>La valeur de cette propriété peut être <code>true</code> ou <code>false</code>. Il permet à un agent d'être configuré pour faire échouer un transfert géré dès qu'un élément de transfert au sein de ce transfert géré échoue.</p> <p>Pour activer cette fonction, l'APAR IT03450 doit être appliqué à la fois à l'agent source et à l'agent cible, et la propriété failTransferOnFirstFailure doit être définie sur <code>true</code> dans le fichier <code>agent.properties</code> de l'agent source. La définition de la propriété sur <code>true</code> sur l'agent de destination est facultative.</p> <p>Lorsque la propriété failTransferOnFirstFailure est définie sur <code>true</code>, l'agent commence à traiter les demandes de transfert gérées normalement. Toutefois, dès qu'un élément de transfert échoue, le transfert géré est marqué comme ayant échoué et aucun autre élément de transfert n'est traité. Les éléments de transfert qui ont été traités avec succès avant l'échec du transfert géré sont traités de la manière suivante:</p> <ul style="list-style-type: none"> • La disposition source de ces éléments de transfert est respectée. Par exemple, si la disposition source de l'élément de transfert a été définie sur <code>delete</code>, le fichier source est supprimé. • Les fichiers de destination qui ont été écrits restent sur le système de fichiers de destination et ne sont pas supprimés. <p>Si la propriété failTransferOnFirstFailure n'est pas définie sur <code>true</code> et qu'un transfert de fichier géré contient plusieurs fichiers et que l'un de ces fichiers ne parvient pas à être transféré, par exemple parce que le fichier de destination existe déjà et que la propriété d'écrasement est définie sur <code>error</code>, l'agent source continue et tente de transférer les fichiers restants dans la demande.</p>	false

Nom de la propriété	Description	Valeur par défaut
itemsPerProgressMessage	<p>Nombre de fichiers transférés avant qu'un agent ne publie son prochain message du journal de progression. Cette propriété contrôle la fréquence à laquelle les messages du journal de progression sont publiés dans le gestionnaire de files d'attente de coordination lors d'un transfert.</p> <p>La valeur maximale sur laquelle cette propriété peut être définie est de 1000.</p> <p>Remarque : Les messages de progression incluent des informations sur chacun des fichiers transférés depuis que le dernier message de progression a été publié. L'augmentation de cette valeur augmente la taille des messages de progression, ce qui peut avoir une incidence sur les performances.</p>	50
maxInlineFileSize	<p>Pour les transferts de fichier à fichier ou de fichier à message, taille de fichier maximale (en octets) pouvant être incluse automatiquement dans le message de demande de transfert initial.</p> <p>Vous pouvez utiliser cette propriété pour accélérer vos transferts, mais si vous spécifiez une taille de fichier trop élevée, cela peut entraîner une dégradation des performances. Il est conseillé d'utiliser une taille initiale de 100 ko pour cette propriété, mais il est recommandé de tester rigoureusement différentes valeurs jusqu'à ce que vous trouviez la taille de fichier qui convient le mieux pour votre système.</p> <p>Cette fonction est désactivée par défaut ou en définissant la propriété maxInlineFileSize sur 0.</p>	0

Nom de la propriété	Description	Valeur par défaut
highlyAvailable	<p>Cette propriété est lue lors du démarrage de l'agent et si elle est définie sur la valeur <code>true</code>, l'agent est démarré en mode haute disponibilité. Si vous ne spécifiez pas la propriété ou si vous définissez la valeur sur <code>false</code>, l'agent démarre en tant qu'agent non hautement disponible.</p>	false
standbyPollIntervalle	<p>Cette propriété est utilisée par l'instance de secours pour tenter d'ouvrir la file d'attente partagée à des intervalles définis.</p> <p>À partir de IBM MQ 9.3.0, cette propriété est également utilisée par toutes les instances pour déterminer la durée pendant laquelle une instance attend entre les tentatives de reconnexion si elle est déconnectée de son gestionnaire de files d'attente d'agent.</p> <p>Les tentatives sont répétées jusqu'à ce qu'une instance se reconnecte à son gestionnaire de files d'attente d'agent, ouvre la file d'attente <code>SYSTEM.FTE.HA.<agent name></code> (si elle s'est déjà enregistrée en tant qu'instance de secours) ou soit arrêtée par la commande <code>fteStopAgent</code>.</p>	5 secondes
standbyStatusDiscardTime	<p>Cette propriété définit la durée pendant laquelle l'instance active attend une publication de statut à partir d'une instance de secours.</p> <p>Si aucune publication n'est reçue d'une instance de secours, même après cette période, l'instance active supprime les informations d'instance de secours de sa liste d'instances de secours.</p> <p>La valeur par défaut est le double de celle de la propriété <code>standbyStatusPublishInterval</code>, de sorte que l'instance active attend plus longtemps avant de supprimer l'instance de secours de sa liste.</p>	600 secondes
standbyStatusExpiration	<p>Cette propriété définit le délai d'expiration du message de statut de secours inséré dans la file d'attente de commandes d'un agent. Le message expire si l'instance active d'un agent ne traite pas ce message.</p>	30 secondes
standbyStatusPublishInterval	<p>Cette propriété permet de définir la fréquence à laquelle l'instance de secours publie son état.</p>	300 secondes

Tableau 74. Propriétés d'agent avancées: Entrée / sortie





Nom de la propriété	Description	Valeur par défaut
doNotUseTempOutputFile	<p>Par défaut, l'agent écrit dans un fichier temporaire sur la destination et renomme ce fichier temporaire avec le nom de fichier requis une fois le transfert de fichier terminé. Si vous définissez cette valeur sur true, l'agent écrit directement dans le fichier de destination final.</p> <p> Sur les systèmes z/OS, ce comportement ne s'applique pas aux fichiers séquentiels, mais il s'applique aux membres des fichiers PDS.</p> <p>La valeur de cette propriété pour un transfert est définie par l'agent de destination.</p>	false
enableMandatoryLocking	<p>Lors de l'accès à des fichiers normaux, Managed File Transfer utilise un verrou partagé pour la lecture et un verrou exclusif pour l'écriture.</p> <p> Sur Windows, le verrouillage de fichier est uniquement une recommandation. Si cette propriété est définie sur true, Managed File Transfer applique le verrouillage des fichiers. Sous Windows, cela signifie que si un fichier est ouvert dans une autre application, la surveillance de ce fichier n'est pas déclenchée tant qu'il n'est pas fermé. Les transferts Managed File Transfer impliquant ce fichier échouent.</p> <p> Sur les plateformes de type UNIX, le verrouillage des fichiers est réalisé entre les processus. Pour les plateformes de type UNIX, la définition de cette propriété n'a aucune incidence.</p> <p>Cette propriété s'applique uniquement aux agents Managed File Transfer normaux. Managed File Transfer ne prend pas en charge le mécanisme de verrouillage de fichier sur un serveur de fichiers. Cette propriété ne fonctionne donc pas pour un agent de pont de protocole car l'agent de pont de protocole ne verrouille pas un fichier sur un serveur de fichiers lors du transfert d'un fichier.</p> <p>La valeur de cette propriété peut être true ou false.</p>	false
ioIdleThreadTimeout	<p>Durée en millisecondes pendant laquelle une unité d'exécution d'entrée/de sortie du système de fichiers doit rester inactive avant d'être arrêtée.</p> <p> Depuis IBM MQ 9.3.0, cette propriété ne s'applique pas aux agents s'exécutant sur IBM MQ for z/OS.</p>	10000
ioQueueDepth	Nombre maximal de demandes d'entrée/de sortie à placer en file d'attente.	10
ioThreadPoolSize	<p>Nombre maximal d'unités d'exécution d'entrée/de sortie du système de fichiers disponibles. En général, chaque transfert utilise sa propre unité d'exécution d'entrée/de sortie de système de fichiers, mais si le nombre de transferts simultanés dépasse cette limite, les unités d'exécution d'entrée/de sortie de système de fichiers sont partagées entre les transferts.</p> <p>Si vous pensez avoir régulièrement plus de transferts simultanés en cours que la valeur ioThreadPoolSize, vous pouvez constater une amélioration en augmentant cette valeur, afin que chaque transfert possède sa propre unité d'exécution d'entrée/de sortie de système de fichiers.</p>	10
textReplacementCharacterSequence	<p>Pour un transfert en mode texte, si des octets de données ne peuvent pas être convertis de la page de codes source vers la page de codes de destination, le transfert de fichier échoue par défaut.</p> <p>Définissez cette propriété pour permettre au transfert de se terminer correctement en insérant la valeur de caractère spécifiée. Cette valeur de propriété est un caractère unique. Généralement, un point d'interrogation (?) est utilisé pour les caractères non mappables. Par exemple, utilisez le format textReplacementCharacterSequence=? où le point d'interrogation (?) est le caractère de remplacement. Vous ne pouvez pas utiliser d'espace blanc comme caractère de remplacement.</p>	Aucun

Tableau 75. Propriétés d'agent avancées: journal de transfert

Nom de la propriété	Description	Valeur par défaut
logTransfer Pour obtenir des exemples des informations de journalisation générées, voir «Sortie générée par la fonction LogTransfer», à la page 224 .	Activez ou désactivez la consignation de transfert. Les valeurs possibles sont les suivantes: informations Active les informations de journal de niveau supérieur d'un transfert. Il s'agit de la valeur par défaut. modérer Active les informations de journal de niveau intermédiaire d'un transfert. prolix Active les informations de journal détaillées d'un transfert. désactivé Désactive la consignation du transfert	informations
logTransferFileSize	Définit la taille maximale d'un fichier journal de transfert en mégaoctets	20
Fichiers logTransfer	Définit le nombre maximal de fichiers de transfert conservés avant que le fichier le plus ancien ne soit supprimé.	5

Tableau 76. Propriétés d'agent avancées: Prise en charge multicanal

Nom de la propriété	Description	Valeur par défaut
agentMultipleChannelsEnabled	La définition de cette propriété sur <code>true</code> permet à un Managed File Transfer Agent d'envoyer des messages de données de transfert sur plusieurs canaux IBM MQ . Dans certains scénarios, la définition de cette propriété peut améliorer les performances. Toutefois, n'activez la prise en charge multicanal que si son incidence positive sur les performances est prouvée. Seuls les messages placés dans la file d'attente SYSTEM.FTE.DATA. <i>nomAgentDestination</i> sont envoyés sur plusieurs canaux. Le comportement de tous les autres messages reste inchangé. Si vous définissez cette propriété sur <code>true</code> , vous devez également effectuer les étapes de configuration d'IBM MQ de l'une des rubriques suivantes pour activer la prise en charge multicanal : <ul style="list-style-type: none"> Configuration d'un agent MFT pour plusieurs canaux dans un cluster Configuration d'un agent MFT pour plusieurs canaux: non configurés en cluster En outre, vous devez également effectuer les étapes de configuration IBM MQ standard requises pour un agent Managed File Transfer , qui sont détaillées dans Configuration de MFT pour la première utilisation . La valeur de cette propriété peut être <code>true</code> ou <code>false</code> .	false
agentMessageBatchSize	S'il est configuré avec plusieurs canaux, un agent source envoie les messages de données d'un transfert sur chaque canal, par permutation circulaire. Cette propriété contrôle le nombre de messages envoyés simultanément par chaque canal.	5

Tableau 77. Propriétés d'agent avancées: Gestionnaire de files d'attente multi-instance

Nom de la propriété	Description	Valeur par défaut
agentQMGrStandby	Nom d'hôte et numéro de port utilisés pour les connexions client, au format CONNAME d'IBM MQ, pour l'instance de secours d'un gestionnaire de files d'attente d'agent multi-instance défini par agentQMGr. Exemple : <code>host_name(port_number)</code> L'agent tente de se connecter au gestionnaire de files d'attente de secours lorsqu'il détecte une erreur de connexion interrompue, par exemple, MQRC 2009. Une fois que l'agent est connecté au gestionnaire de files d'attente de secours, il reste connecté jusqu'à ce que le gestionnaire de files d'attente de secours devienne indisponible.	Aucune valeur par défaut

Tableau 78. Propriétés d'agent avancées: Contrôleur de processus

Nom de la propriété	Description	Valeur par défaut
agentQMGrRetryInterval	Intervalle, en secondes, entre les vérifications de disponibilité du gestionnaire de files d'attente par le contrôleur de processus de l'agent.	30

Tableau 78. Propriétés d'agent avancées: Contrôleur de processus (suite)		
Nom de la propriété	Description	Valeur par défaut
maxRestartCount	Nombre maximal de redémarrages pouvant se produire dans l'intervalle de temps spécifié par la valeur de la propriété maxRestartInterval. Si cette valeur est dépassée, le contrôleur de processus de l'agent arrête le redémarrage de l'agent et effectue à la place une action qui dépend de la valeur de la propriété maxRestartDelay.	4
maxRestartInterval	Intervalle, en secondes, au cours duquel le contrôleur de processus de l'agent mesure les redémarrages de ce dernier. Si le nombre de redémarrages au cours de cet intervalle dépasse la valeur de la propriété maxRestartCount, le contrôleur de processus de l'agent arrête le redémarrage de l'agent. Au lieu de cela, il effectue une action qui dépend de la valeur de la propriété maxRestartDelay.	120
maxRestartDelay	Détermine le comportement du contrôleur de processus de l'agent si la fréquence des redémarrages de l'agent dépasse la valeur des propriétés maxRestartCount et maxRestartInterval. Si vous spécifiez une valeur inférieure ou égale à zéro, le contrôleur de processus de l'agent est arrêté. Si vous spécifiez une valeur supérieure à zéro, cette dernière correspond au nombre de secondes à attendre avant que les informations sur l'historique des redémarrages conservées par le contrôleur de processus de l'agent ne soient réinitialisées et que l'agent ne soit redémarré.	-1

Tableau 79. Propriétés d'agent avancées: pont de protocole		
Nom de la propriété	Description	Valeur par défaut
protocolBridgeCredentialConfiguration	La valeur de cette propriété est transmise comme chaîne à la méthode initialize() des classes d'exit spécifiées par protocolBridgeCredentialExitClasses.	null
protocolBridgeCredentialExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des données d'identification du pont de protocole. Pour plus d'informations, voir Mapping credentials for a file server by using exit classes .	Aucune valeur par défaut
protocolBridgeDataTimeout	Délai, en millisecondes, avant que l'agent de pont de protocole n'établisse une connexion de données avec un serveur FTP ou ne reçoive des données d'un serveur FTP par l'intermédiaire d'une connexion déjà établie. Si vous définissez cette propriété sur la valeur 0, l'agent de pont de protocole attend indéfiniment. Une fois le délai écoulé, l'agent de pont de protocole ferme toutes les connexions de données existantes au serveur FTP et tente d'établir une nouvelle connexion de données avant de reprendre le transfert en cours. Si la tentative visant à établir la nouvelle connexion de données échoue, le transfert en cours échoue également.	0
protocolBridgeLogoutBeforeDisconnect	Indique si l'agent de pont de protocole déconnecte l'utilisateur du serveur de fichiers avant de fermer la session FTP et de se déconnecter. Si vous définissez cette propriété sur true, l'agent de pont de protocole envoie une commande FTP QUIT au serveur de fichiers.	false
protocolBridgePropertiesConfiguration	Transmise comme l'une des propriétés de pont à la méthode initialize() des classes d'exit spécifiées par la propriété protocolBridgeServerPropertiesExitClasses.	Aucune valeur par défaut
protocolBridgePropertiesExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des propriétés serveur du pont de protocole. Pour plus d'informations, voir ProtocolBridgePropertiesExit2: Looking up protocol file server properties .	Aucune valeur par défaut

Tableau 80. Propriétés d'agent avancées: consignation de l'agent de pont de protocole		
Nom de la propriété	Description	Valeur par défaut
agentLog	Composant de la paire clé/valeur et opération permettant d'activer ou de désactiver la consignation des commandes FTP et des réponses entre l'agent de pont de protocole et les serveurs de fichiers FTP/SFTP/FTPS. Exemple : agentLog=on Active la consignation pour tous les composants agentLog=off Désactive la consignation pour tous les composants agentLog=ftp=on, sftp=on, ftps=off Active la consignation pour FTP et SFTP et la désactive pour FTPS	Aucune valeur par défaut
agentLogFileSize	Définit la taille maximale d'un fichier de capture en mégaoctets. Identique à la valeur par défaut de la taille de fichier standard par défaut de la trace.	20

Tableau 80. Propriétés d'agent avancées: consignation de l'agent de pont de protocole (suite)

Nom de la propriété	Description	Valeur par défaut
agentLogFiles	<p>Définit le nombre maximal de fichiers de capture conservés avant que le fichier le plus ancien ne soit supprimé.</p> <ul style="list-style-type: none"> La valeur par défaut de la propriété d'agent agentLogFiles est passée de 10 à 5. Cela signifie qu'à partir de IBM MQ 9.3.0, si la valeur par défaut est définie, il peut y avoir un maximum de cinq fichiers journaux d'événements d'agent de pont de protocole, en commençant par <code>agentevent0.log</code> à <code>agentevent4.log</code>. Toutefois, vous pouvez modifier cette valeur si nécessaire. Si l'agent est migré à partir d'une version antérieure à IBM MQ 9.3.0, vous devez supprimer manuellement les fichiers de <code>agentevent5.log</code> à <code>agentevent9.log</code>, le cas échéant. Cependant, la taille de chaque fichier journal reste de 20 Mo. 	Depuis IBM MQ 9.3.0, la valeur par défaut est 5.
agentLogFilter	<p>Par défaut, capture les communications avec tous les serveurs FTP auxquels l'agent se connecte.</p> <p>Exemple :</p> <ul style="list-style-type: none"> Filtrage sur le nom d'hôte/l'adresse IP <pre>host=ftpprod.ibm.com, ftp2.ibm.com host=9.182.*</pre> <ul style="list-style-type: none"> Filtrage basé sur les métadonnées <pre>metadata="outbound files to xyz corp"</pre>	*

Tableau 81. Propriétés d'agent avancées: File d'attente

Nom de la propriété	Description	Valeur par défaut
Préfixe dynamicQueue	Cette propriété définit le préfixe à utiliser lors de la création d'une file d'attente dynamique temporaire.	WMQFTE.*
modelQueueNom	Cette propriété définit le nom de la file d'attente de module à utiliser lors de la création d'une file d'attente dynamique temporaire.	SYSTEM.DEFAULT.MODEL.QUEUE
publicationMDUser	ID utilisateur MQMD à associer aux messages envoyés pour être publiés par le gestionnaire de file d'attente de coordination. Si vous ne définissez pas cette propriété, l'ID utilisateur MQMD est défini en fonction des règles IBM MQ de définition des ID utilisateur MQMD.	Aucune valeur par défaut

Tableau 82. Propriétés d'agent avancées: Surveillance des ressources

Nom de la propriété	Description	Valeur par défaut
monitorFilepathPlatformSeparator	Indique si des séparateurs de chemin spécifiques à la plateforme doivent être utilisés dans la variable <code>\$FILEPATH</code> . La valeur <code>true</code> utilise des séparateurs de chemin spécifiques à la plateforme. La valeur <code>false</code> utilise un séparateur de chemin de style UNIX (<code>/</code>) sur toutes les plateformes.	Oui
monitorMaxResourcesInPoll	<p>Indique le nombre maximal de ressources surveillées à déclencher lors de chaque intervalle d'interrogation. Par exemple, si vous spécifiez le modèle de surveillance <code>*.txt</code> et un intervalle d'interrogation de 10 secondes et que vous définissez la propriété <code>monitorMaxResourcesInPoll</code> sur 10, la propriété <code>monitorMaxResourcesInPoll</code> limite les déclenchements de l'agent à 10 correspondances par intervalle d'interrogation. Les ressources correspondantes au delà de la limite de 10 sont déclenchées dans les intervalles d'interrogation ultérieurs.</p> <p>En outre, vous pouvez utiliser la propriété <code>monitorMaxResourcesInPoll</code> en combinaison avec un paramètre <code>-bs</code> correspondant sur la commande fteCreateMonitor, par exemple, pour restreindre chaque intervalle d'interrogation au déclenchement d'un seul transfert.</p> <p>Si vous spécifiez une valeur inférieure ou égale à zéro, le nombre de ressources de surveillance déclenchées dans un intervalle d'interrogation est illimité.</p>	-1
monitorReportTriggerFail	Indique si les conditions d'échec, dans l'environnement et la configuration, détectées dans le moniteur, sont envoyées sous forme de message de journal à la rubrique <code>SYSTEM.FTE</code> . La valeur <code>true</code> consigne les messages. La valeur <code>false</code> ne consigne pas les messages.	Oui

Tableau 82. Propriétés d'agent avancées: Surveillance des ressources (suite)

Nom de la propriété	Description	Valeur par défaut
monitorReportTriggerNotSatisfied	Indique si un déclencheur non satisfait envoie un message de journal à la rubrique SYSTEM.FTE qui contient les détails. La valeur <code>true</code> consigne les messages. La valeur <code>false</code> ne consigne pas les messages.	false
monitorReportTriggerSatisfied	Indique si un déclencheur satisfait envoie un message de journal à la rubrique SYSTEM.FTE qui contient les détails. La valeur <code>true</code> consigne les messages. La valeur <code>false</code> ne consigne pas les messages.	false
monitorSilenceOnTriggerFailure	Le nombre d'échecs consécutifs du déclencheur du moniteur de ressources avant les échecs n'est plus signalé.	5
monitorStopOnInternalFailure	Nombre de conditions FFDC internes consécutives du moniteur de ressources avant que le moniteur ne passe à l'état d'arrêt.	10

Tableau 83. Propriétés avancées de l'agent: répertoire racine

Nom de la propriété	Description	Valeur par défaut
commandPath	<p>Spécifie l'ensemble des chemins d'accès par lesquels les commandes peuvent être appelées, à l'aide de l'une des méthodes suivantes :</p> <ul style="list-style-type: none"> Tâches Ant fte: <code>call Ant task</code>, fte: <code>filecopyyou fte: filemove</code> de l'agent Dans un message XML transmis à un agent, un des schémas XML de commande Managed File Transfer Agent pris en charge (par exemple, <code>managedCall</code> ou <code>managedTransfer</code>). <p>Pour plus d'informations sur la syntaxe valide de la valeur de la propriété <code>commandPath</code>, voir commandPath MFT property.</p> <p>Important : Soyez extrêmement prudent lorsque vous définissez cette propriété car toute commande de l'un des <code>commandPath</code> spécifiés peut être appelée efficacement à partir d'un système client distant capable d'envoyer des commandes à l'agent. Pour cette raison, par défaut, lorsque vous spécifiez un <code>commandPath</code>:</p> <ul style="list-style-type: none"> Tout bac à sable d'agent existant est configuré par l'agent au démarrage afin que tous les répertoires <code>commandPath</code> soient automatiquement ajoutés à la liste des répertoires auxquels l'accès est refusé pour un transfert. Tous les bacs à sable d'utilisateur existants sont mis à jour au démarrage de l'agent de sorte que tous les répertoires <code>commandPath</code> (et leurs sous-répertoires) soient ajoutés en tant qu'éléments <code><exclude></code> aux éléments <code><read></code> et <code><write></code>. Si l'agent n'est pas configuré pour utiliser un bac à sable d'agent ou des bacs à sable d'utilisateur, un nouveau bac à sable d'agent est créé au démarrage de l'agent avec les répertoires <code>commandPath</code> spécifiés comme étant des répertoires refusés. <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.</p> <p>Vous pouvez définir la propriété <code>addCommandPathToSandbox</code> sur <code>false</code> pour remplacer ce comportement par défaut.</p> <p>Important : Sachez que ce remplacement permet à un client de transférer toute commande vers le système de l'agent et d'appeler la commande et doit donc être utilisé avec précaution.</p>	Aucune - Aucune commande ne peut être appelée
addCommandPathToSandbox	<p>Indique si les répertoires spécifiés par la propriété <code>commandPath</code> (et tous leurs sous-répertoires) doivent être ajoutés à:</p> <ul style="list-style-type: none"> Aux répertoires refusés d'un bac à sable d'agent existant. Les éléments <code><exclude></code> pour les éléments <code><read></code> et <code><write></code> pour les bacs à sable utilisateur qui ont été définis. A un nouveau bac à sable d'agent, si aucun agent n'a été configuré avec un bac à sable d'agent ou un ou plusieurs bacs à sable d'utilisateur. <p>Pour plus d'informations sur la syntaxe valide de la valeur de la propriété <code>commandPath</code>, voir commandPath MFT property.</p>	True

Tableau 83. Propriétés avancées de l'agent: répertoire racine (suite)

Nom de la propriété	Description	Valeur par défaut
additionalWildcardSandboxChecking	<p>Indique si des vérifications supplémentaires doivent être effectuées sur les transferts de caractères génériques pour un agent qui a été configuré avec un bac à sable d'utilisateur ou d'agent afin de restreindre les emplacements vers et depuis lesquels l'agent peut transférer des fichiers.</p> <p>Lorsque cette propriété est définie sur true, la vérification supplémentaire est activée. Si une demande de transfert tente de lire un emplacement qui se trouve en dehors du bac à sable défini pour la correspondance de fichier du caractère générique, le transfert échoue. S'il existe plusieurs transferts dans une même demande de transfert et que l'une de ces demandes échoue en raison d'une tentative de lecture d'un emplacement en dehors du bac à sable, le transfert complet échoue. Si la vérification échoue, la cause de l'échec est indiquée dans des messages d'erreur (voir Contrôles supplémentaires pour les transferts de caractères génériques).</p> <p>Si la propriété est omise ou définie sur false, aucune vérification supplémentaire n'est effectuée sur les transferts de caractères génériques.</p>	Aucun
sandboxRoot	<p>Spécifie l'ensemble des chemins d'accès racine à inclure et exclure lorsque vous utilisez les bacs à sable. Pour plus d'informations sur cette fonction, voir Utilisation des bacs à sable de l'agent MFT.</p> <p>Séparez les chemins d'accès par des séparateurs de chemin d'accès spécifiques à la plateforme. Préfixez les chemins avec un point d'exclamation (!) pour indiquer que les chemins sont exclus du bac à sable. Cette fonction est utile si vous souhaitez exclure un sous-répertoire sous un chemin d'accès racine inclus.</p> <p>La propriété sandboxRoot n'est pas prise en charge sur les agents de pont de protocole.</p> <p>Vous ne pouvez pas spécifier les propriétés sandboxRoot et userSandboxes ensemble.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.</p>	Aucune - Pas de bac à sable
transferRoot	<p>Répertoire racine par défaut pour les chemins d'accès relatifs indiqués à l'agent.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.</p>	Répertoire de base de l'utilisateur ayant démarré le processus de l'agent.
transferRootHLQ	Qualificatif de haut niveau (ID utilisateur) par défaut pour les fichiers non complets spécifiés à l'agent	Nom d'utilisateur de l'utilisateur ayant démarré le processus de l'agent.
userSandboxes	<p>Restreint la zone du système de fichiers vers et depuis laquelle les fichiers peuvent être transférés en fonction du nom utilisateur MQMD de l'utilisateur qui demande le transfert. Pour plus d'informations, voir Utilisation des bacs à sable utilisateur MFT.</p> <p>La propriété userSandboxes n'est pas prise en charge sur les agents de pont de protocole.</p> <p>Vous ne pouvez pas spécifier les propriétés sandboxRoot et userSandboxes ensemble.</p>	false

Tableau 84. Propriétés d'agent avancées: propriété Planificateur

Nom de la propriété	Description	Valeur par défaut
maxSchedulerRunDelay	<p>Délai d'attente maximal, en minutes, avant que l'agent ne recherche les transferts planifiés. Spécifiez un entier positif pour activer cette propriété. Pour plus d'informations sur la raison pour laquelle vous pouvez utiliser cette propriété, voir Que faire si votre transfert de fichiers planifié ne s'exécute pas ou est retardé.</p> <p>L'agent pouvant lire une commande de sa file d'attente de commandes au moment où les transferts planifiés doivent être exécutés, un délai supplémentaire peut s'écouler avant que ces transferts planifiés ne démarrent. Dans ce cas, le planificateur est exécuté dès que cette commande est terminée.</p>	-1

Tableau 85. Propriétés d'agent avancées: sécurité







Nom de la propriété	Description	Valeur par défaut
agentCredentialsKeyFile	Nom du fichier contenant la clé de données d'identification utilisée lors du chiffrement des données d'identification.	Propriété de chaîne sans valeur par défaut.
agentQMGrAuthenticationCredentialsFile	<p>Chemin d'accès au fichier contenant les données d'identification à utiliser lors de la connexion au gestionnaire de files d'attente de l'agent.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.</p> <p> Pour plus de détails sur la création des fichiers de données d'identification, voir MFT and IBM MQ connection authentication.</p> <p> Pour plus de détails sur la création du fichier de données d'authentification, voir Configuration de MQMFTCredentials.xml sur z/OS.</p>	<p>La valeur par défaut de cette propriété est:</p> <p> z/OS</p> <p> Linux</p> <p> AIX \$HOME/MQMFTCredentials.xml</p> <p> Windows %USERPROFILE%/MQMFTCredentials.xml</p>
authorityChecking	<p>Indique si les fonctions de sécurité décrites dans Restriction des droits utilisateur sur les actions de l'agent MFT sont activées.</p> <p>Le droit inquire est requis dans toutes les files d'attente d'autorisations d'agent.</p>	false
logAuthorityChecks	Niveau de consignation de la vérification des droits enregistré dans le journal des événements d'agent, dans le fichier output0.log. Les valeurs possibles pour cette propriété sont None, Failures et All.	Aucun
userIdForClientConnect	ID utilisateur transmis par l'intermédiaire des connexions client à IBM MQ. Si java est spécifié, le nom d'utilisateur signalé par la machine virtuelle Java est transmise dans le cadre de la demande de connexion à IBM MQ. Les valeurs possibles pour cette propriété sont None et java.	Aucun

Tableau 86. Propriétés d'agent avancées: SSL/TLS


Nom de la propriété	Description	Valeur par défaut
agentSslCipherSpec	<p>Indique le protocole, l'algorithme de hachage et l'algorithme de chiffrement utilisés, ainsi que le nombre de bits utilisés dans la clé de chiffrement, lorsque les données sont échangées entre l'agent et le gestionnaire de files d'attente de l'agent.</p> <p>La valeur de agentSslCipherSpec est un nom CipherSpec. Ce nom CipherSpec est identique au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente de l'agent. La liste des noms CipherSpec valides est incluse dans les CipherSpecs SSL/TLS CipherSpecs et CipherSuites dans les classes IBM MQ pour Java et SSL/TLS CipherSpecs et CipherSuites dans les classes IBM MQ pour JMS.</p> <p>agentSslCipherSpec est similaire à agentSslCipherSuite. Si agentSslCipherSuite et agentSslCipherSpec sont spécifiés, la valeur de agentSslCipherSpec est utilisée.</p>	Aucun
agentSslCipherSuite	<p>Indique les aspects SSL de la façon dont l'agent et le gestionnaire de files d'attente de l'agent échangent des données.</p> <p>La valeur de agentSslCipherSuite est un nom CipherSuite. Le nom CipherSuite est mappé au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente de l'agent. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings.</p> <p>agentSslCipherSuite est similaire à agentSslCipherSpec. Si agentSslCipherSuite et agentSslCipherSpec sont spécifiés, la valeur de agentSslCipherSpec est utilisée.</p>	Aucun
agentSslPeerName	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente d'agent. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente lors de la connexion.	Aucun
agentSslTrustStore	<p>Indique l'emplacement des certificats auxquels l'agent fait confiance. La valeur de agentSslTrustStore est un chemin de fichier. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappées à l'aide d'une barre oblique inversée (\).</p> <p> Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/).</p> <p>La valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucun

Tableau 86. Propriétés d'agent avancées: SSL/TLS (suite)


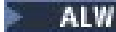











Nom de la propriété	Description	Valeur par défaut
agentSslKeyStore	Indique l'emplacement de la clé privée de l'agent. La valeur de agentSslKeyStore est un chemin de fichier. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappées à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Cette propriété est requise uniquement si le gestionnaire de files d'attente de l'agent requiert l'authentification du client. La valeur de cette propriété peut contenir des variables d'environnement.	Aucun
agentSslFipsRequired	Indique que vous souhaitez activer la prise en charge FIPS au niveau de l'agent. La valeur de cette propriété peut être true ou false. Pour plus d'informations, voir Prise en charge de FIPS dans MFT .	false
agentSslKeyStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
agentSslKeyStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification permettant d'accéder au magasin de clés de l'agent. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.  Pour plus de détails sur la création des fichiers de données d'identification, voir MFT and IBM MQ connection authentication .  Pour plus de détails sur la création du fichier de données d'authentification, voir Configuration de MQMFTCredentials.xml sur z/OS .	La valeur par défaut de cette propriété est:    \$HOME/MQMFTCredentials.xml  %USERPROFILE%/MQMFTCredentials.xml
agentSslTrustStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
agentSslTrustStoreCredentialsFile	Chemin d'accès au fichier contenant les données d'identification permettant d'accéder au magasin de clés de confiance de l'agent. La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.  Pour plus de détails sur la création des fichiers de données d'identification, voir MFT and IBM MQ connection authentication .  Pour plus de détails sur la création du fichier de données d'authentification, voir Configuration de MQMFTCredentials.xml sur z/OS .	La valeur par défaut de cette propriété est:    \$HOME/MQMFTCredentials.xml  %USERPROFILE%/MQMFTCredentials.xml

Tableau 87. Propriétés d'agent avancées: Délai d'attente

Nom de la propriété	Description	Valeur par défaut
maxTransferNegotiationTime	Délai maximal pendant lequel un transfert attend qu'un agent de destination termine la négociation. Si la négociation ne se termine pas dans le délai imparti, le transfert est placé dans un état de resynchronisation, ce qui permet l'exécution d'un autre transfert, quand ce dernier est disponible. Dans les scénarios où l'agent source ou de destination est soumis à une charge élevée, il se peut que la valeur par défaut soit trop faible pour que l'agent réponde assez rapidement à la demande de négociation. Ce cas se produit généralement lorsqu'un nombre important de moniteurs de ressources sont définis sur un agent source ou lorsque les moniteurs de ressources de cet agent surveillent des répertoires qui contiennent un grand nombre de fichiers. Il peut toutefois se produire quand un nombre important de demandes de transfert sont soumises à un agent. Il peut alors s'avérer nécessaire de spécifier une valeur supérieure ou égale à 200 000 pour cette propriété.	30 000
recoverableTransferRetryInterval	Délai d'attente en millisecondes entre la détection d'une erreur de transfert récupérable et une tentative de reprise du transfert.	60 000

Nom de la propriété	Description	Valeur par défaut
senderTransferRetryInterval	Délai d'attente en millisecondes avant qu'un transfert rejeté soit relancé car la destination exécute déjà le nombre maximal de transferts. La valeur minimale est de 1 000.	30 000
transferAckTimeout	Délai en millisecondes pendant lequel un transfert attend l'accusé de réception ou des données de l'autre extrémité avant d'effectuer une nouvelle tentative. Il s'agit d'une propriété avancée et pour la plupart des configurations Managed File Transfer, il n'est pas nécessaire d'en modifier la valeur. Les accusés de réception sont envoyés par l'agent réceptionnaire à l'agent émetteur chaque fois qu'une fenêtre complète de données est reçue. Pour les réseaux non fiables ou dont la bande passante est limitée et les paramètres agentWindowSize et agentChunkSize dont la valeur est élevée, il est possible que la valeur par défaut ne suffise pas. Cela peut entraîner de nouveaux transferts inutiles entre les agents. Par conséquent, l'augmentation de cette valeur peut être bénéfique et permettre de réduire la probabilité du passage d'un transfert en mode reprise en raison d'un réseau lent.	60 000
transferAckTimeoutRetries	Nombre maximal de nouvelles tentatives d'accusé de réception pour un transfert sans obtenir de réponse avant que l'agent n'abandonne et ne place le transfert à l'état de récupération	5
xmlConfigReloadInterval	Intervalle en secondes entre les rechargements par l'agent des fichiers de configuration XML lors de la phase d'exécution. Pour empêcher l'agent de recharger les fichiers de configuration XML pendant la phase d'exécution, définissez cette propriété sur -1. Les fichiers de configuration XML suivants sont affectés par cette propriété : <ul style="list-style-type: none"> • ConnectDirectCredentials.xml • ConnectDirectNodeProperties.xml • ConnectDirectProcessDefinitions.xml • ProtocolBridgeCredentials.xml • ProtocolBridgeProperties.xml • UserSandboxes.xml 	30

Nom de la propriété	Description	Valeur par défaut
javaCoreTriggerFile	Chemin d'accès complet à un emplacement de fichier surveillé par l'agent. Si le fichier existe à l'emplacement indiqué, le démarrage de l'agent déclenche un Javacore. Après avoir démarré l'agent, si vous mettez à jour un fichier à cet emplacement, l'agent déclenche à nouveau un fichier Javacore. Une unité d'exécution distincte interroge ce fichier toutes les 30 secondes pour déterminer si le fichier a été créé ou mis à jour. Si le fichier a été créé ou mis à jour depuis la dernière interrogation, l'agent génère un fichier Javacore dans le répertoire suivant: <code>MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/agents/agent_name</code> Si vous spécifiez cette propriété, l'agent génère le message suivant au démarrage : <pre>BFGAG0092I The <insert_0> file will be used to request JVM diagnostic information.</pre> La valeur de cette propriété peut contenir des variables d'environnement. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.	Aucun

Tableau 88. Propriétés d'agent avancées: traçage et journalisation (suite)

Nom de la propriété	Description	Valeur par défaut
trace	<p>Spécification de trace à utiliser lorsque l'agent est démarré. Il s'agit d'une liste de classes et / ou de packages séparés par des virgules, du caractère égal et d'un niveau de trace.</p> <p>Par exemple, pour tracer la classe <code>com.ibm.wmqfte.agent.Agent</code> et les classes du package <code>com.ibm.wmqfte.commandhandler</code> à partir du démarrage de l'agent, ajoutez l'entrée suivante au fichier <code>agent.properties</code> :</p> <pre>trace=com.ibm.wmqfte.agent.Agent,com.ibm.wmqfte.commandhandler=all</pre> <p>Vous pouvez spécifier plusieurs spécifications de trace dans une liste de valeurs séparées par deux-points. Exemple :</p> <pre>trace=com.ibm.wmqfte.agent.Agent=all:com.ibm.wmqfte.commandhandler=moderate</pre> <p>La spécification de trace spéciale <code>=all</code> est utilisée pour tracer l'agent et l'interface JMQUI (Java Message Queuing Interface) qui gère toutes les communications avec le gestionnaire de files d'attente de l'agent. Pour l'activer, ajoutez l'entrée suivante au fichier <code>agent.properties</code> :</p> <pre>trace==all</pre> <p>Sauf indication contraire de votre représentant de support IBM, utilisez la spécification de trace <code>com.ibm.wmqfte=all</code> comme suit:</p> <pre>trace=com.ibm.wmqfte=all</pre>	Aucun
outputLogFiles	Nombre total de fichiers <code>output.log</code> à conserver. Cette valeur s'applique au contrôleur de processus d'un agent et à l'agent lui-même.	5
outputLogSize	Taille maximale en Mo de chaque fichier <code>output.log</code> avant que la sortie ne passe au prochain fichier. Cette valeur s'applique au contrôleur de processus d'un agent et à l'agent lui-même.	1
outputLogEncoding	Codage de caractères utilisé par l'agent lorsqu'il écrit dans le fichier <code>output.log</code> .	Codage de caractères par défaut de la plateforme sur laquelle l'agent est exécuté.
traceFiles	Nombre total de fichiers de trace à conserver. Cette valeur s'applique au contrôleur de processus d'un agent, ainsi qu'à l'agent lui-même.	5
traceSize	Taille maximale en Mo de chaque fichier de trace avant que la trace ne passe au prochain fichier. Cette valeur s'applique au contrôleur de processus d'un agent et à l'agent lui-même.	20
traceMaxBytes	Quantité maximale de données de message générée dans le fichier de trace.	4096 octets
logTransferRecovery	Lorsque cette propriété est définie sur la valeur <code>true</code> , chaque fois qu'un transfert entre des événements de diagnostic de reprise est signalé dans le journal des événements de l'agent dans le fichier <code>output0.log</code> .	La valeur par défaut est <code>true</code> .
logCapture	Capture les messages de demande de transfert soumis à cet agent et les messages de journal publiés par l'agent sur le gestionnaire de file d'attente de coordination. Ces messages capturés peuvent s'avérer utiles lors du débogage des problèmes de transfert. Les messages capturés sont stockés dans des fichiers, dans le répertoire de journaux d'agent appelé <code>capture?.log</code> . ? est une valeur numérique. Le fichier qui contient le chiffre 0 contient les messages capturés les plus récents.	<code>false</code>
logCaptureFileSize	Définit la taille maximale d'un fichier de capture en mégaoctets.	10
logCaptureFiles	Définit le nombre maximal de fichiers de capture conservés avant que le fichier le plus ancien ne soit supprimé.	10
logCaptureFilter	Expression régulière Java utilisée par l'agent pour correspondre au nom de rubrique du message. Seuls les messages correspondant à l'expression régulière sont capturés.	* (correspondance totale)

Tableau 88. Propriétés d'agent avancées: traçage et journalisation (suite)


Nom de la propriété	Description	Valeur par défaut
Journal resourceMonitor	<p>Paire clé-valeur du moniteur de ressources et de l'opération pour activer ou désactiver la journalisation.</p> <p>Les valeurs possibles sont les suivantes:</p> <ul style="list-style-type: none"> informations modérer prolix désactivé <p>Exemple :</p> <ul style="list-style-type: none"> resourceMonitorLog=MON1,MON2=info:MON3=off <p>Activez la journalisation pour MON1 et MON2 et désactivez la journalisation pour MON3.</p> <ul style="list-style-type: none"> resourceMonitorLog=info <p>Activez la journalisation de niveau info pour tous les moniteurs de ressources.</p> <p>Les journaux du moniteur de ressources sont écrits dans un fichier nommé resmoneventN.log, où N représente un nombre ; par exemple, resmonevent0.log.</p> <p> Avertissement : Tous les moniteurs de ressources d'un agent écrivent dans le même fichier journal.</p> <p>Pour plus d'informations, voir Journalisation des moniteurs de ressources MFT.</p>	informations
Taille de resourceMonitorLogFile	Définit la taille maximale d'un fichier de capture en mégaoctets.	20
resourceMonitorLogFiles	<p>Définit le nombre maximal de fichiers de capture conservés avant que le fichier le plus ancien ne soit supprimé.</p> <ul style="list-style-type: none"> La valeur par défaut de la propriété d'agent resourceMonitorLogFiles est passée de 10 à 5. Cela signifie que, à partir de IBM MQ 9.3.0, si la valeur par défaut est définie, il peut y avoir un maximum de cinq fichiers journaux d'événements du moniteur de ressources, à partir de resmonevent0.log vers resmonevent4.log. Toutefois, vous pouvez modifier cette valeur si nécessaire. Si l'agent est migré à partir d'une version antérieure à IBM MQ 9.3.0, vous devez supprimer manuellement les fichiers de resmonevent5.log à resmonevent9.log, le cas échéant. Cependant, la taille de chaque fichier journal reste de 20 Mo. 	Depuis IBM MQ 9.3.0, la valeur par défaut est 5.

Tableau 89. Propriétés d'agent avancées: limite de transfert

Nom de la propriété	Description	Valeur par défaut
maxDestinationTransfers	<p>Nombre maximal de transferts simultanés traités par l'agent de destination à tout moment. Chaque demande de transfert soumise à un agent est comptabilisée dans ce total, quel que soit le nombre de fichiers transférés pour satisfaire la demande. Cela signifie qu'une demande de transfert qui transfère un fichier unique compte autant qu'une demande de transfert qui transfère 10 fichiers.</p> <p>L'agent place les transferts en file d'attente lorsque l'agent de destination a atteint la limite spécifiée par la propriété maxDestinationTransfers.</p> <p>Si la somme des valeurs des propriétés maxSourceTransfers, maxDestinationTransfers et maxQueuedTransfers dépasse la valeur du paramètre MAXDEPTH de la file d'attente du magasin d'état (SYSTEM.FTE.STATE.nom_agent), l'agent ne démarre pas.</p>	<p>25 (pour tous les agents à l'exception de Connect:Direct)</p> <p>5 (pour les agents de pont Connect:Direct)</p>
maxFilesForTransfer	<p>Nombre maximal d'éléments de transfert autorisés pour un transfert géré unique. Si un transfert géré contient plus d'éléments que la valeur de maxFilesForTransfer, le transfert géré échoue et aucun élément de transfert n'est traité.</p> <p>La définition de cette propriété vous empêche de transférer accidentellement un trop grand nombre de fichiers en raison d'une demande de transfert incorrecte, par exemple, si un utilisateur spécifie accidentellement le transfert du répertoire racine / sur le mot clé conref=" ../common/mqent.dita#mqent/unixlinuxbis " />.</p>	5000

Nom de la propriété	Description	Valeur par défaut
maxSourceTransfers	<p>Nombre maximal de transferts simultanés traités par l'agent source à tout moment. Chaque demande de transfert soumise à un agent est comptabilisée dans ce total, quel que soit le nombre de fichiers transférés pour satisfaire la demande. Cela signifie qu'une demande de transfert qui transfère un fichier unique compte autant qu'une demande de transfert qui transfère 10 fichiers.</p> <p>L'agent source place les transferts en file d'attente lorsque l'agent de destination a atteint la limite spécifiée par la propriété maxSourceTransfers.</p> <p>Si la somme des valeurs des propriétés maxSourceTransfers, maxDestinationTransfers et maxQueuedTransfers dépasse la valeur du paramètre MAXDEPTH de la file d'attente du magasin d'état (SYSTEM.FTE.STATE.nom_agent), l'agent ne démarre pas.</p>	<p>25 (pour tous les agents, excepté les agents de pont Connect:Direct)</p> <p>5 (pour les agents de pont Connect:Direct)</p>
maxQueuedTransfers	<p>Nombre maximal de transferts en attente pouvant être mis en file d'attente par un agent source jusqu'à ce que l'agent rejette une nouvelle demande de transfert. Vous pouvez définir cette propriété de sorte que même si les limites des propriétés maxDestinationTransfers et maxSourceTransfers sont atteintes ou dépassées, les nouvelles demandes de transfert que vous effectuez maintenant sont acceptées, mises en file d'attente, puis réalisées ultérieurement.</p> <p>L'ordre suivant lequel les demandes de transfert en file d'attente sont traitées dépend de leur priorité et du temps qu'elles sont restées en file d'attente. Les transferts en attente anciens et à priorité élevée sont sélectionnés en premier. Les transferts à priorité basse qui sont restés longtemps dans la file d'attente sont sélectionnés de préférence par rapport aux transferts à priorité élevée plus récents.</p> <p>Si la somme des valeurs des propriétés maxSourceTransfers, maxDestinationTransfers et maxQueuedTransfers dépasse la valeur du paramètre MAXDEPTH de la file d'attente du magasin d'état (SYSTEM.FTE.STATE.nom_agent), l'agent ne démarre pas.</p>	1000

Nom de la propriété	Description	Valeur par défaut
transferRecoveryTimeout	<p>Délai, en secondes, au cours duquel un agent source continue de tenter de restaurer un transfert de fichier bloqué.</p> <p>Si cette propriété n'est pas définie, par défaut, l'agent effectue de nouvelles tentatives jusqu'à ce que la récupération du transfert aboutisse. Vous pouvez définir les valeurs suivantes pour la propriété de délai de reprise des transferts :</p> <p>-1 L'agent continue de tenter de reprendre le transfert bloqué jusqu'à ce que ce dernier soit terminé. Cette option est équivalente au comportement par défaut de l'agent si la propriété n'est pas définie.</p> <p>0 L'agent arrête le transfert de fichiers dès qu'il entre dans la phase de reprise.</p> <p>>0 L'agent continue de tenter de reprendre le transfert bloqué pendant le nombre de secondes défini par l'entier positif spécifié. Par exemple, transferRecoveryTimeout=21600 indique que l'agent continue de tenter de reprendre le transfert pendant six heures à partir du moment où ce dernier est entré en phase de récupération. La valeur maximale de ce paramètre est 999999999.</p>	-1

Nom de la propriété	Description	Valeur par défaut
agentForceConsistentPathDelimiters	Forcé le délimiteur de chemin dans les informations de fichier source et de fichier de destination fournies aux exits de transfert pour qu'il soit de type UNIX : barre oblique (/). Les options valides sont true et false.	false
destinationTransferEndExitClasses	Indique une liste de classes séparées par des virgules qui implémentent une routine d'exit utilisateur de transfert de destination.	Aucune valeur par défaut
destinationTransferStartExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur de début de transfert de destination.	Aucune valeur par défaut

Tableau 91. Propriétés d'agent avancées: routine d'exit utilisateur (suite)		
Nom de la propriété	Description	Valeur par défaut
exitClassPath	Indique une liste de répertoires, délimités par des caractères, spécifique à la plateforme, qui servent de chemins d'accès aux classes des routines d'exit utilisateur. Le répertoire des exits de l'agent est recherché avant toutes les entrées de ce chemin d'accès aux classes.	Répertoire exits de l'agent
exitNativeLibraryPath	Indique une liste de répertoires, délimités par des caractères, spécifique à la plateforme, qui servent de chemins d'accès aux bibliothèques natives des routines d'exit utilisateur.	Répertoire exits de l'agent
ioMaxRecordLength	Longueur maximale d'un enregistrement, en octets, pouvant être prise en charge pour un fichier orienté enregistrements. Managed File Transfer peut prendre en charge l'écriture dans des fichiers orientés enregistrement avec toute longueur d'enregistrement. Toutefois, des longueurs d'enregistrement importantes peuvent entraîner des erreurs de mémoire insuffisante. Par conséquent, pour éviter ces erreurs, la longueur d'enregistrement maximale est limitée par défaut à 64 ko. Lors de la lecture à partir de fichiers orientés enregistrement, un enregistrement entier doit tenir dans un seul bloc de transfert, de sorte que la longueur de l'enregistrement est en outre limitée par la taille du bloc de transfert. Cette propriété n'est utilisée que pour les fichiers orientés enregistrement des existes utilisateur d'E-S.	64 ko
monitorExitClasses	Spécifie une liste de classes implémentant une routine d'exit de moniteur, séparées par des virgules. Pour plus d'informations, voir Exits utilisateur du moniteur de ressources MFT .	Aucune valeur par défaut
protocolBridgeCredentialExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur des données d'identification du pont de protocole. Pour plus d'informations, voir Mapping credentials for a file server by using exit classes .	Aucune valeur par défaut
sourceTransferEndExitClasses	Spécifie une liste de classes implémentant une routine d'exit de fin de transfert source.	Aucune valeur par défaut
sourceTransferStartExitClasses	Spécifie une liste de classes implémentant une routine d'exit de début de transfert source.	Aucune valeur par défaut
IOExitClasses	Spécifie une liste de classes implémentant une routine d'exit utilisateur d'E-S, séparées par des virgules. Ne répertorie que les classes qui implémentent l'interface IOExit et non celles qui implémentent d'autres interfaces d'exit utilisateur d'E-S, comme IOExitResourcePath et IOExitChannel. Pour plus d'informations, voir Using MFT transfer I/O user exits .	Aucune valeur par défaut

Tableau 92. Propriétés d'agent avancées: compression du client IBM MQ		
Nom de la propriété	Description	Valeur par défaut
agentDataCompression	Cette propriété n'est prise en charge que pour les connexions client. Liste de types de compression, séparés par des virgules, pour le transfert de données de fichier à négocier avec le serveur IBM MQ distant. Pour plus d'informations sur ces types de compression, reportez-vous à la rubrique suivante : Liste de compression des données de message . Les valeurs sont vérifiées pour vérifier leur validité, puis transmises par ordre d'apparence en tant que propriétés au canal client de l'agent. Le client IBM MQ gère ensuite la négociation entre ce canal client et le canal du serveur distant pour rechercher le plus petit dénominateur commun correspondant entre les propriétés de compression sur les deux canaux. Si aucune correspondance n'est trouvée, MQCOMPRESS_NONE est toujours sélectionné.	MQCOMPRESS_NONE
agentHeaderCompression	Cette propriété n'est prise en charge que pour les connexions client. Liste de types de compression, séparés par des virgules, pour le transfert de données d'en-tête à négocier avec le serveur IBM MQ distant. Les valeurs admises sont MQCOMPRESS_NONE et MQCOMPRESS_SYSTEM. Vous trouverez des informations sur ces types de compression dans la rubrique suivante: HdrCompList [2] (MQLONG) . Les valeurs sont vérifiées pour vérifier leur validité, puis transmises par ordre d'apparence en tant que propriétés au canal client de l'agent. Le client IBM MQ gère ensuite la négociation entre ce canal client et le canal du serveur distant pour rechercher le plus petit dénominateur commun correspondant entre les propriétés de compression sur les deux canaux. Si aucune correspondance n'est trouvée, MQCOMPRESS_NONE est toujours sélectionné.	MQCOMPRESS_NONE

Tableau 93. Propriétés d'agent avancées: z/OS-specific


Nom de la propriété	Description	Valeur par défaut
adminGroup	<p>Un groupe de gestionnaires de sécurité. Les membres de ce groupe peuvent :</p> <ul style="list-style-type: none"> Démarrer l'agent à l'aide de la commande fteStartAgent. Arrêter l'agent à l'aide de la commande fteStopAgent. Activez ou désactivez la trace de l'agent à l'aide de la commande fteSetAgentTraceLevel. Activez ou désactivez les journaux de l'agent à l'aide de la commande fteSetAgentLogLevel. Affichez les détails d'un agent local en exécutant la commande fteShowAgentDetails avec le paramètre -d spécifié. <p>Définissez un groupe de gestionnaires de sécurité, par exemple MFTADMIN, puis ajoutez l'ID utilisateur de la tâche démarrée et les ID TSO de l'administrateur à ce groupe. Editez le fichier de propriétés de l'agent et définissez la propriété adminGroup pour qu'elle soit le nom de ce groupe de gestionnaires de sécurité.</p> <pre>adminGroup=MFTADMIN</pre>	Aucun
 bpxwdynAllocAdditionalOptions	<p>Managed File Transfer utilise l'interface textuelle BPXWDYN pour créer et ouvrir des fichiers z/OS. Si BPXWDYN est utilisé pour l'allocation de fichier, par défaut, Managed File Transfer s'assure, si possible, que l'unité de données est montée (non requis pour les fichiers sur disque, mais requis pour les fichiers sur bande). Les options pouvant ne pas être prises en charge pour certains environnements, utilisez cette propriété pour modifier ce comportement. En outre, lors du transfert vers un fichier, il est également possible de spécifier des options pour BPXWDYN sur la ligne de commande ; ces options complètent celles spécifiées par cette propriété.</p> <p>Certaines options BPXWDYN ne doivent pas être spécifiées lors de l'utilisation de la propriété bpxwdynAllocAdditionalOptions dans le fichier <code>agent.properties</code>. Pour la liste de ces propriétés, voir Propriétés BPXWDYN que vous ne devez pas utiliser avec MFT.</p>	<p>La valeur par défaut se présente comme suit :</p> <ul style="list-style-type: none"> MOUNT pour z/OS V1R8 et versions ultérieures
armELEMTYPE	Propriété facultative. Si l'agent est configuré pour être redémarré par ARM (Automatic Restart Manager), définissez cette propriété sur la valeur du paramètre ARM ELEMTYPE spécifiée dans la règle ARM associée. Pour un agent, définissez ELEMTYPE sur SYSBFGAG.	Non définie
armELEMENT	Propriété facultative. Si l'agent est configuré pour être redémarré par ARM (Automatic Restart Manager), définissez cette propriété sur la valeur du paramètre ARM ELEMENT spécifiée dans la règle ARM associée. Vous pouvez définir la valeur ELEMENT de sorte qu'elle corresponde au nom de l'agent.	Non définie

Tableau 94. Propriétés d'agent avancées: autres propriétés



Nom de la propriété	Description	Valeur par défaut
  legacyXMLMessageMQMDFormat	<p>Les messages XML de Managed File Transfer générés par l'agent (par exemple, les messages de journal et les messages de progression du transfert), sont désormais envoyés à une file d'attente avec une zone vide au format MQMD. Les versions précédentes du produit définissaient la zone au format MQMD sur MQSTR (chaîne de message texte). La définition de cette propriété sur true permet d'envoyer les messages XML de Managed File Transfer générés par l'agent à une file d'attente avec une zone au format MQMD définie sur MQSTR.</p> <p>Remarque : les messages de réponse de l'agent aux commandes seront envoyés avec un format de message correspondant à la demande de commande correspondante.</p> <p>Si la zone au format MQMD est définie sur MQSTR, les messages XML des commandes Managed File Transfer risquent d'être corrompus si la conversion des données est activée pour des canaux du réseau MQ.</p>	false

Tableau 94. Propriétés d'agent avancées: autres propriétés (suite)

Nom de la propriété	Description	Valeur par défaut
adjustScheduleTimeForDaylightSavings	<p>Si votre entreprise exécute des transferts planifiés tous les jours, car le transfert planifié a été créé avec:</p> <ul style="list-style-type: none"> Paramètre -oi défini sur les jours, et Paramètre -tb défini sur la source <p>sur la commande <code>fteCreateTransfer</code> par exemple, si vous affectez à cette propriété la valeur <code>true</code>, l'heure de transfert planifiée est avancée d'une heure lorsque les horloges sont avancées d'une heure et d'une heure lorsque les horloges sont reculées d'une heure.</p> <p>Par exemple, si votre transfert planifié doit être exécuté à 1h00, lorsque les horloges sont avancées, le transfert est exécuté à 2h00 et lorsque les horloges sont reculées, le transfert est rétabli à 1h00.</p>	Oui

Concepts associés

[Options de configuration de MFT sur Multiplatforms](#)

[Option de délai d'attente pour les transferts de fichiers en cours de récupération](#)

[MFT bacs à sable](#)

Tâches associées

[Configuration d'un agent MFT pour plusieurs canaux dans un cluster](#)

[Configuration d'un agent MFT pour plusieurs canaux: non configurés en cluster](#)

Référence associée

«Propriétés système Java pour MFT», à la page 227

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java, car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

[Propriétés SSL/TLS pour MFT](#)

«Le fichier MFT `command.properties`», à la page 210

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

«Le fichier MFT `coordination.properties`», à la page 205

Le fichier `coordination.properties` indique les détails de connexion au gestionnaire de file d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier `coordination.properties` commun sur une unité partagée.

«Le fichier MFT `logger.properties`», à la page 214

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

[fteCreateAgent](#)

[fteCreateBridgeAgent](#)

[fteCreateCDAgent](#)

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 175

Il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction de l'environnement en cours (par exemple, l'utilisateur exécutant une commande).

Le fichier MFT `coordination.properties`

Le fichier `coordination.properties` indique les détails de connexion au gestionnaire de file d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier `coordination.properties` commun sur une unité partagée.

Le fichier `coordination.properties` est créé par le programme d'installation ou par la commande **fteSetupCoordination**. Vous pouvez utiliser la commande **fteSetupCoordination** avec l'indicateur **-f** pour modifier les propriétés de base du gestionnaire de files d'attente de coordination dans ce fichier. Pour modifier ou ajouter des propriétés de gestionnaire de files d'attente de coordination avancées, vous devez éditer le fichier dans un éditeur de texte.

Le fichier `coordination.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`.

Le fichier MFT `coordination.properties` contient les valeurs suivantes:

Tableau 95. Propriétés de base du gestionnaire de files d'attente de coordination		
Nom de la propriété	Description	Valeur par défaut
<code>coordinationCredentialsKeyFile</code>	Nom du fichier contenant la clé de données d'identification utilisée lors du chiffrement des données d'identification.	Propriété de chaîne sans valeur par défaut.
<code>coordinationQMGr</code>	Nom du gestionnaire de file d'attente de coordination.	Aucune valeur par défaut
Hôte <code>coordinationQMGr</code>	Nom d'hôte ou adresse IP du gestionnaire de file d'attente de coordination.	Aucune valeur par défaut
Port <code>coordinationQMGr</code>	Numéro de port utilisé pour les connexions client au gestionnaire de file d'attente de coordination.	1414
Canal <code>coordinationQMGr</code>	Nom de canal SVRCONN utilisé pour la connexion au gestionnaire de file d'attente de coordination.	SYSTEM.DEF.SVRCONN

Si vous ne spécifiez pas de valeur pour la propriété d'hôte `coordinationQMGr`, le mode de liaison est utilisé par défaut.


Si vous spécifiez une valeur pour la propriété d'hôte `coordinationQMGr` mais que vous ne spécifiez pas de valeurs pour les propriétés de port `coordinationQMGr` et de canal `coordinationQMGr`, le numéro de port 1414 et le canal `SYSTEM.DEF.SVRCONN` sont utilisés par défaut.

Voici un exemple du contenu d'un fichier `coordination.properties` :

```
coordinationQMGr=ERIS
coordinationQMGrHost=kuiper.example.com
coordinationQMGrPort=2005
coordinationQMGrChannel=SYSTEM.DEF.SVRCONN
```

Dans cet exemple, ERIS est le nom d'un gestionnaire de files d'attente IBM MQ situé sur le système `kuiper.example.com`. Le gestionnaire de files d'attente ERIS est le gestionnaire de files d'attente auquel Managed File Transfer envoie les informations de journal.

Propriétés de coordination avancées

Managed File Transfer fournit également des propriétés de coordination plus avancées. Si vous souhaitez utiliser l'une des propriétés suivantes, éditez manuellement le fichier `coordination.properties` pour ajouter les propriétés avancées requises. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Pour plus d'informations sur la mise en échappement des caractères dans les fichiers de propriétés Java, voir la documentation Oracle [Javadoc pour la classe Properties](#).

- [Propriétés d'agent](#)
- [Propriétés de page de codes](#)
- [Propriétés de connexion](#)
- [Propriétés du gestionnaire de files d'attente multi-instance](#)
- [Propriétés de file d'attente](#)
- [Propriétés de sécurité](#)
- [propriétés SSL](#)

• Propriétés d'abonnement

Tableau 96. Propriétés de coordination avancées: Agent

Nom de la propriété	Description	Valeur par défaut
agentStatusJitterTolerance	<p>Durée maximale pendant laquelle une publication de message d'état d'agent peut être retardée avant que le message ne soit considéré comme étant en retard. Cette valeur est mesurée en millisecondes.</p> <p>L'âge d'un message de statut dépend de l'heure à laquelle il a été publié au niveau du gestionnaire de files d'attente de coordination. Toutefois, le message est émis par l'agent un certain temps avant d'être reçu au niveau du gestionnaire de file d'attente de coordination afin de permettre le temps nécessaire pour parcourir le réseau IBM MQ . Si ce transit prend toujours le même temps, les messages créés à 60 secondes d'intervalle sont publiés à 60 secondes d'intervalle, quelle que soit la durée réelle du transit. Cependant, si le temps de transit varie entre les messages, ils peuvent être créés à des intervalles de 60 secondes mais publiés à des intervalles de 61, 59, 58 et 62 secondes par exemple. L'écart maximal de 60, 2 secondes dans cet exemple, est la gigue. Cette propriété détermine le délai maximal dû à la gigue avant que le message soit traité comme étant en retard.</p>	3000

Tableau 97. Propriétés de coordination avancées: page de codes

Nom de la propriété	Description	Valeur par défaut
coordinationCcsid	Page de codes avec laquelle les commandes se connectent au gestionnaire de file d'attente de coordination. Toutes les publications du gestionnaire de files d'attente de coordination effectuées par l'agent sont également effectuées à l'aide de cette page de codes. Si vous spécifiez une valeur pour <code>coordinationCcsid</code> , vous devez également spécifier une valeur pour <code>coordinationCcsid</code> .	1208
coordinationCcsidNom	Représentation Java de <code>coordinationCcsid</code> . Si vous spécifiez une valeur pour le nom <code>coordinationCcsid</code> , vous devez également spécifier une valeur pour <code>coordinationCcsid</code> .	UTF8

Tableau 98. Propriétés de coordination avancées: Connexion

Nom de la propriété	Description	Valeur par défaut
javaLibraryPath	Lors de la connexion à un gestionnaire de files d'attente en mode liaisons, Managed File Transfer doit avoir accès aux bibliothèques de liaisons IBM MQ Java . Par défaut, Managed File Transfer recherche les bibliothèques de liaisons dans l'emplacement par défaut défini par IBM MQ. Si les bibliothèques de liaisons se trouvent à un autre emplacement, utilisez cette propriété pour spécifier l'emplacement des bibliothèques de liaisons.	<code>MQ_INSTALLATION_PATH/java/lib</code>

Tableau 99. Propriétés de coordination avancées: Gestionnaire de files d'attente multi-instance

Nom de la propriété	Description	Valeur par défaut
coordinationQMgrde secours	Nom d'hôte et numéro de port utilisés pour les connexions client, au format IBM MQ CONNAME, pour l'instance de secours d'un gestionnaire de files d'attente de coordination multi-instance défini par la propriété <code>coordinationQMgr</code> . Exemple : <code>host_name(port_number)</code>	Aucune valeur par défaut

Tableau 100. Propriétés de coordination avancées: File d'attente

Nom de la propriété	Description	Valeur par défaut
Préfixe dynamicQueue	<p>Cette propriété définit le préfixe IBM MQ à utiliser pour générer un nom de file d'attente temporaire.</p> <p>Le format de la propriété de préfixe <code>dynamicQueues</code> suit le format de la zone DynamicQName de la structure IBM MQ MQOD. Pour plus d'informations, voir Création de files d'attente dynamiques.</p> <p>Vous pouvez également définir cette propriété dans le fichier <code>command.properties</code> si vous souhaitez utiliser un préfixe IBM MQ spécifique pour les files d'attente de réponses temporaires générées par des commandes nécessitant une réponse de l'agent.</p>	WMQFTE.*

Tableau 100. Propriétés de coordination avancées: File d'attente (suite)		
Nom de la propriété	Description	Valeur par défaut
modelQueueNom	<p>Cette propriété définit la file d'attente modèle IBM MQ à utiliser pour générer une file d'attente temporaire.</p> <p>Vous pouvez également définir cette propriété dans le fichier <code>command.properties</code> si vous souhaitez utiliser une file d'attente modèle IBM MQ spécifique pour les files d'attente de réponses temporaires générées par des commandes nécessitant une réponse de l'agent. Pour plus d'informations, voir «Le fichier MFT <code>command.properties</code>», à la page 210.</p>	SYSTEM.DEFAULT.MODEL.QUEUE






Tableau 101. Propriétés de coordination avancées: sécurité		
Nom de la propriété	Description	Valeur par défaut
userIdForClientConnect	ID utilisateur transmis par l'intermédiaire des connexions client à IBM MQ. Si <code>java</code> est spécifié, le nom d'utilisateur signalé par la machine virtuelle Java est transmis dans le cadre de la demande de connexion IBM MQ. La valeur de cette propriété peut être <code>None</code> ou <code>java</code> .	Aucun
coordinationQMGrAuthenticationCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de coordination.	<p> Pour plus de détails sur la création du fichier de données d'authentification, voir Configuration de MQMFTCredentials.xml sur z/OS.</p> <p> Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml.</p> <p> Pour plus de détails sur la création du fichier de données d'identification d'authentification, voir MFT et IBM MQ authentification de la connexion.</p>

Tableau 102. Propriétés de coordination avancées: SSL/TLS		
Nom de la propriété	Description	Valeur par défaut
coordinationSslCipherSpec	<p>Indique le protocole, l'algorithme de hachage et l'algorithme de chiffrement utilisés, ainsi que le nombre de bits utilisés dans la clé de chiffrement, lorsque les données sont échangées entre les commandes et le gestionnaire de files d'attente de coordination.</p> <p>La valeur de <code>coordinationSslCipherSpec</code> est un nom <code>CipherSpec</code>. Ce nom <code>CipherSpec</code> est identique au nom <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de coordination. La liste des noms <code>CipherSpec</code> valides est incluse dans les CipherSpecs SSL/TLS CipherSpecs et CipherSuites dans les classes IBM MQ pour Java et SSL/TLS CipherSpecs et CipherSuites dans les classes IBM MQ pour JMS.</p> <p><code>coordinationSslCipherSpec</code> est similaire à <code>coordinationSslCipherSuite</code>. Si <code>coordinationSslCipherSuite</code> et <code>coordinationSslCipherSpec</code> sont spécifiés, la valeur de <code>coordinationSslCipherSpec</code> est utilisée.</p>	Aucun
coordinationSslCipherSuite	<p>Indique les aspects SSL de la façon dont les commandes et le gestionnaire de files d'attente de coordination échangent des données.</p> <p>La valeur de <code>coordinationSslCipherSuite</code> est un nom <code>CipherSuite</code>. Le nom <code>CipherSuite</code> est mappé au nom <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de l'agent. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings.</p> <p><code>coordinationSslCipherSuite</code> est similaire à <code>coordinationSslCipherSpec</code>. Si <code>coordinationSslCipherSuite</code> et <code>coordinationSslCipherSpec</code> sont spécifiés, la valeur de <code>coordinationSslCipherSpec</code> est utilisée.</p>	Aucun
coordinationSslPeerName	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente de coordination. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente de coordination lors de la connexion.	Aucun

Nom de la propriété	Description	Valeur par défaut
coordinationSslTrustStore	Indique l'emplacement des certificats auxquels les commandes font confiance. La valeur de <code>coordinationSslTrustStore</code> est un chemin de fichier. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Depuis la IBM WebSphere MQ 7.5 ou une version ultérieure, la valeur de cette propriété peut contenir des variables d'environnement.	Aucun
coordinationSslTrustStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>
coordinationSslTrustStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification <code>coordinationSslTrustStore</code> . La valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est <code>%USERPROFILE%/MQMFTCredentials.xml</code> sur Windows et <code>\$HOME/MQMFTCredentials.xml</code> sur d'autres plateformes.
coordinationSslKeyStore	Indique l'emplacement de la clé privée des commandes. La valeur de <code>coordinationSslKeyStore</code> est un chemin de fichier. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Cette propriété est requise uniquement si le gestionnaire de files d'attente de coordination requiert une authentification de client. La valeur de cette propriété peut contenir des variables d'environnement.	Aucun
coordinationSslKeyStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>
coordinationSslKeyStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification <code>coordinationSslKeyStore</code> . La valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est <code>%USERPROFILE%/MQMFTCredentials.xml</code> sur Windows et <code>\$HOME/MQMFTCredentials.xml</code> sur d'autres plateformes.
coordinationSslFipsRequired	Indique que vous souhaitez activer la prise en charge FIPS au niveau du gestionnaire de file d'attente de coordination. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> . Pour plus d'informations, voir Prise en charge de FIPS dans MFT .	<code>false</code>

Nom de la propriété	Description	Valeur par défaut
Rubrique <code>coordinationSubscription</code>	Utilisez cette propriété pour spécifier une rubrique autre que <code>SYSTEM.FTE</code> auquel s'abonner pour obtenir des publications sur le statut du réseau IBM MQ. Tous les outils sont toujours publiés dans <code>SYSTEM.FTE</code> , mais vous pouvez modifier votre topologie IBM MQ pour distribuer ces publications dans différentes rubriques en fonction de leur contenu. Vous pouvez ensuite utiliser cette fonction pour forcer les outils à s'abonner à l'une de ces autres rubriques.	<code>SYSTEM.FTE</code>

Concepts associés

Options de configuration de MFT sur Multiplatforms

Référence associée

[fteSetupCoordination](#)

[Propriétés SSL/TLS pour MFT](#)

«Le fichier `MFT.agent.properties`», à la page 181

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente.

Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

«Le fichier MFT `command.properties`», à la page 210

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

«Le fichier MFT `logger.properties`», à la page 214

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Le fichier MFT `command.properties`

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

Le fichier `command.properties` est créé par le programme d'installation ou par la commande **`fteSetupCommands`**. Vous pouvez utiliser la commande **`fteSetupCommands`** avec l'indicateur **`-f`** pour modifier les propriétés de base du gestionnaire de files d'attente de commandes dans ce fichier. Pour modifier ou ajouter des propriétés de gestionnaire de files d'attente de commandes avancées, vous devez éditer le fichier dans un éditeur de texte.

Certaines commandes Managed File Transfer se connectent au gestionnaire de files d'attente d'agent ou au gestionnaire de files d'attente de coordination à la place du gestionnaire de files d'attente de commandes. Pour plus d'informations sur les commandes qui se connectent à un gestionnaire de files d'attente, voir [Quelle commande MFT se connecte à quel gestionnaire de files d'attente](#).

Le fichier `command.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name`.

Le fichier MFT `command.properties` contient les valeurs suivantes:

Nom de la propriété	Description	Valeur par défaut
<code>connectionCredentialsKeyFile</code>	Nom du fichier contenant la clé de données d'identification utilisée lors du chiffrement des données d'identification.	Propriété de chaîne sans valeur par défaut.
<code>connectionQMGr</code>	Nom du gestionnaire de files d'attente utilisé pour la connexion au réseau IBM MQ.	Aucune valeur par défaut
Hôte <code>connectionQMGr</code>	Nom d'hôte ou adresse IP du gestionnaire de files d'attente de connexion.	Aucune valeur par défaut
<code>connectionQMGrPort</code>	Numéro de port utilisé pour la connexion au gestionnaire de files d'attente de connexion en mode client.	1414
Canal <code>connectionQMGr</code>	Nom de canal SVRCONN utilisé pour la connexion au gestionnaire de files d'attente de connexion.	SYSTEM.DEF.SVRCONN

Si vous ne spécifiez pas de valeur pour la propriété d'hôte `connectionQMGr`, le mode de liaison est utilisé par défaut.


Si vous spécifiez une valeur pour la propriété d'hôte `connectionQMGr` mais que vous ne spécifiez pas de valeurs pour les propriétés de port `connectionQMGrPort` et de canal `connectionQMGrChannel`, le numéro de port 1414 et le canal `SYSTEM.DEF.SVRCONN` sont utilisés par défaut.

Voici un exemple du contenu d'un fichier `command.properties` :

```
connectionQMGr=PLUTO
connectionQMGrHost=kuiper.example.com
connectionQMGrPort=1930
connectionQMGrChannel=SYSTEM.DEF.SVRCONN
```

Dans cet exemple, PLUTO est le nom d'un gestionnaire de files d'attente IBM MQ qui se trouve sur le système `kuiper.example.com`. Le gestionnaire de files d'attente PLUTO est le gestionnaire de files d'attente auquel les commandes Managed File Transfer se connectent.

Propriétés de commande avancées

Managed File Transfer fournit également des propriétés de commande plus avancées. Si vous souhaitez utiliser l'une des propriétés suivantes, éditez manuellement le fichier `command.properties` pour ajouter les propriétés avancées requises. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Pour plus d'informations sur la mise en échappement des caractères dans les fichiers de propriétés Java, voir la documentation Oracle [Javadoc](#) pour la classe `Properties`.

- [Propriétés d'agent](#)
- [Propriétés de page de codes](#)
- [Propriétés du gestionnaire de files d'attente multi-instance](#)
- [Propriétés de file d'attente](#)
- [Propriétés de sécurité](#)
- [propriétés SSL](#)

Tableau 105. Propriétés de commande avancées: Agent

Nom de la propriété	Description	Valeur par défaut
<code>failCleanAgentWithNoArguments</code>	Par défaut, la valeur de cette propriété est true, ce qui signifie que l'exécution de la commande fteCleanAgent échoue si seul le paramètre de nom d'agent est spécifié. La définition de la propriété sur false signifie que, si seul le paramètre de nom d'agent est défini, le comportement de la commande fteCleanAgent est équivalent à la spécification du paramètre -all .	Oui

Tableau 106. Propriétés de commande avancées: page de codes

Nom de la propriété	Description	Valeur par défaut
<code>connectionCcsid</code>	Page de codes avec laquelle les commandes se connectent au gestionnaire de files d'attente de commandes. Si vous spécifiez une valeur pour <code>connectionCcsid</code> , vous devez également spécifier une valeur pour le nom <code>connectionCcsid</code> .	1208
<code>connectionCcsidNom</code>	Représentation Java de <code>connectionCcsid</code> . Si vous spécifiez une valeur pour le nom <code>connectionCcsid</code> , vous devez également spécifier une valeur pour <code>connectionCcsid</code> .	UTF8

Tableau 107. Propriétés de connexion avancées: gestionnaire de files d'attente multi-instance

Nom de la propriété	Description	Valeur par défaut
<code>connectionQMgrde secours</code>	Nom d'hôte et numéro de port utilisés pour les connexions client, au format IBM MQ CONNAME, pour l'instance de secours d'un gestionnaire de files d'attente de commandes multi-instance défini par la propriété <code>connectionQMgr</code> . Exemple : <code>host_name(port_number)</code>	Aucune valeur par défaut

Tableau 108. Propriétés de commande avancées: File d'attente

Nom de la propriété	Description	Valeur par défaut
Préfixe <code>dynamicQueue</code>	Pour les commandes qui requièrent une réponse de l'agent, cette propriété définit le préfixe IBM MQ à utiliser pour générer le nom de la file d'attente de réponses temporaire. Le format de la propriété de préfixe <code>dynamicQueue</code> suit le format de la zone DynamicQName de la structure IBM MQ MQOD. Pour plus d'informations, voir Création de files d'attente dynamiques . Vous pouvez également définir cette propriété dans le fichier <code>coordination.properties</code> si vous souhaitez utiliser un préfixe IBM MQ spécifique pour les files d'attente temporaires générées par WMQFTE.	WMQFTE.*





Tableau 108. Propriétés de commande avancées: File d'attente (suite)		
Nom de la propriété	Description	Valeur par défaut
modelQueueNom	Pour les commandes qui requièrent une réponse de l'agent, cette propriété définit la file d'attente modèle IBM MQ à utiliser pour la génération de la file d'attente de réponses temporaire. Vous pouvez également définir cette propriété dans le fichier <code>coordination.properties</code> si vous souhaitez utiliser une file d'attente modèle IBM MQ spécifique pour les files d'attente temporaires générées par WMQFTE. Pour plus d'informations, voir «Le fichier MFT <code>coordination.properties</code> », à la page 205.	SYSTEM.DEFAULT.MODEL.QUEUE
Propriétés de connexion :		
javaLibraryPath	Lors de la connexion à un gestionnaire de files d'attente en mode liaisons, Managed File Transfer doit avoir accès aux bibliothèques de liaisons IBM MQ Java . Par défaut, Managed File Transfer recherche les bibliothèques de liaisons dans l'emplacement par défaut défini par IBM MQ. Si les bibliothèques de liaisons se trouvent à un autre emplacement, utilisez cette propriété pour spécifier l'emplacement des bibliothèques de liaisons.	/opt/mqm/java/lib
  legacyXMLMessageMQMDFormat	Les messages XML de commande Managed File Transfer sont désormais envoyés à une file d'attente avec une zone de format MQMD vide. Les versions précédentes du produit définissaient la zone au format MQMD sur MQSTR (chaîne de message texte). La définition de cette propriété sur true permet d'envoyer les messages XML de la commande Managed File Transfer à une file d'attente avec la zone de format MQMD de MQSTR. Si la zone au format MQMD est définie sur MQSTR, les messages XML des commandes Managed File Transfer risquent d'être corrompus si la conversion des données est activée pour des canaux du réseau MQ.	false

Tableau 109. Propriétés de commande avancées: Sécurité		
Nom de la propriété	Description	Valeur par défaut
userIdForClientConnect	ID utilisateur transmis par l'intermédiaire des connexions client à IBM MQ. Si <i>java</i> est spécifié, le nom d'utilisateur signalé par la machine virtuelle Java est transmis dans le cadre de la demande de connexion IBM MQ . La valeur de cette propriété peut être None ou java.	Aucun
connectionQMGrAuthenticationCredentialsFichier	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de commandes.	Voir MFT and IBM MQ connection authentication et ses rubriques enfant.

Tableau 110. Propriétés de commande avancées: SSL/TLS		
Nom de la propriété	Description	Valeur par défaut
connectionSslCipherSpec	Indique le protocole, l'algorithme de hachage et l'algorithme de chiffrement utilisés, ainsi que le nombre de bits utilisés dans la clé de chiffrement, lorsque les données sont échangées entre les commandes et le gestionnaire de files d'attente de commandes. La valeur de <code>connectionSslCipherSpec</code> est un nom <code>CipherSpec</code> . Ce nom de <code>CipherSpec</code> est identique au nom de <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de commandes. Une liste de noms de <code>CipherSpec</code> valides est incluse dans les <code>CipherSpecsSSL/TLS CipherSpecs</code> et les <code>CipherSuites</code> dans les classes IBM MQ pour Java et les <code>CipherSpecs CipherSpecs</code> et les <code>CipherSuites</code> dans les classes IBM MQ pour JMS. <code>connectionSslCipherSpec</code> est similaire à <code>connectionSslCipherSuite</code> . Si <code>connectionSslCipherSuite</code> et <code>connectionSslCipherSpec</code> sont spécifiés, la valeur de <code>connectionSslCipherSpec</code> est utilisée.	Aucun
connectionSslCipherSuite	Indique les aspects SSL de la façon dont les commandes et le gestionnaire de files d'attente de commandes échangent des données. La valeur de <code>connectionSslCipherSuite</code> est un nom <code>CipherSuite</code> . Le nom <code>CipherSuite</code> est mappé au nom <code>CipherSpec</code> utilisé sur le canal du gestionnaire de files d'attente de l'agent. Pour plus d'informations, voir CipherSuite et CipherSpec name mappings . <code>connectionSslCipherSuite</code> est similaire à <code>connectionSslCipherSpec</code> . Si <code>connectionSslCipherSuite</code> et <code>connectionSslCipherSpec</code> sont spécifiés, la valeur de <code>connectionSslCipherSpec</code> est utilisée.	Aucun
connectionSslPeerName	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente de commandes. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente de commandes lors de la connexion.	Aucun

Tableau 110. Propriétés de commande avancées: SSL/TLS (suite)

Nom de la propriété	Description	Valeur par défaut
connectionSslTrustStore	Indique l'emplacement des certificats auxquels les commandes font confiance. La valeur de connectionSslTrustStore est un chemin de fichier. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). La valeur de cette propriété peut contenir des variables d'environnement.	Aucun
connectionSslTrustStoreType	Type de magasin de clés de confiance SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12.	jks
connectionSslTrustStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification connectionSslTrustStore . La valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est %USERPROFILE%/MQMFTcredentials.xml sur Windows et \$HOME/MQMFTcredentials.xml sur d'autres plateformes.
connectionSslKeyStore	Indique l'emplacement de la clé privée des commandes. La valeur de connectionSslKeyStore est un chemin de fichier. Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).  Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Cette propriété est requise uniquement si le gestionnaire de files d'attente de commandes requiert l'authentification du client. La valeur de cette propriété peut contenir des variables d'environnement.	Aucun
connectionSslKeyStore	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être jks ou pkcs12. La valeur de cette propriété peut contenir des variables d'environnement.	jks
connectionSslKeyStoreCredentialsFile	Chemin d'accès au fichier qui contient les données d'identification connectionSslKeyStore . La valeur de cette propriété peut contenir des variables d'environnement.	La valeur par défaut de cette propriété est %USERPROFILE%/MQMFTcredentials.xml sur Windows et \$HOME/MQMFTcredentials.xml sur d'autres plateformes.
connectionSslFipsRequired	Indique que vous souhaitez activer la prise en charge FIPS au niveau du gestionnaire de files d'attente de commandes. La valeur de cette propriété peut être true ou false. Pour plus d'informations, voir Prise en charge de FIPS dans MFT .	false

Concepts associés

[Options de configuration de MFT sur Multiplatforms](#)

Référence associée

«Propriétés système Java pour MFT», à la page 227

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java , car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

[Propriétés SSL/TLS pour MFT](#)

«Le fichier MFT agent.properties», à la page 181

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, agent.properties, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier agent.properties peut également contenir des propriétés qui modifient le comportement de l'agent.

«Le fichier MFT coordination.properties», à la page 205

Le fichier coordination.properties indique les détails de connexion au gestionnaire de file d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même

gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier `coordination.properties` commun sur une unité partagée.

«Le fichier `MFT logger.properties`», à la page 214

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Commandes `fteSetup`: créez le fichier `MFT command.properties`

Agent `fteClean`: nettoyage d'un agent MFT

Le fichier `MFT logger.properties`

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

Vous pouvez utiliser des variables d'environnement dans certaines propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction des changements d'environnement, tels que l'utilisateur qui exécute le processus. Pour plus d'informations, voir «Utilisation des variables d'environnement dans les propriétés MFT», à la page 175.

Remarque : Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappés à l'aide d'une barre oblique inversée (\).

Windows Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/). Pour plus d'informations sur l'échappement des caractères dans les fichiers de propriétés Java dans Oracle, voir Javadoc pour la classe `Properties`.

Le fichier `MFT logger.properties` contient les valeurs suivantes:

- «Propriétés de connexion en mode liaisons», à la page 214
- «Propriétés de connexion SSL/TLS en mode client», à la page 222

Propriétés de connexion en mode liaisons

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.logger.type</code>	Type de consignateur utilisé: fichier ou base de données. Définissez cette valeur sur <code>FILE</code> ou <code>DATABASE</code> .	Aucune valeur par défaut
<code>wmqfte.max.transaction.messages</code>	Nombre maximal de messages traités dans une transaction avant la validation de la transaction. En mode de journalisation avec réutilisation automatique des journaux, un gestionnaire de files d'attente dispose d'une quantité fixe d'espace disponible pour les données en cours. Veillez à définir cette propriété avec une valeur suffisamment faible pour que l'espace disponible ne soit pas saturé.	50
<code>wmqfte.max.transaction.time</code>	Durée maximale, en millisecondes, qui s'écoule entre les validations de transaction.	5000

Tableau 111. Propriétés de connexion en mode liaison pour le fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.max.consecutive.reject</code>	<p>Nombre maximal de messages pouvant être rejetés consécutivement (c'est-à-dire sans rencontrer de message valide).</p> <p>Si ce nombre est dépassé, le consignateur conclut que le problème n'est pas lié aux messages eux-mêmes, mais à la configuration. Par exemple, si vous créez une colonne de nom d'agent dans la base de données plus étroite que tous les noms d'agent, tous les messages faisant référence aux agents sont rejetés.</p>	50
<code>wmqfte.reject.queue.name</code>	Nom d'une file d'attente dans laquelle le consignateur insère des messages qu'il ne peut pas traiter. Si vous disposez d'un consignateur de base de données, voir Traitement et rejet des erreurs du consignateurMFT pour plus de détails sur les messages pouvant être placés dans cette file d'attente.	SYSTEME SYSTEM.FTE.LOG.RJCT. <i>nom_connexion</i>
<code>wmqfte.command.queue.name</code>	Nom d'une file d'attente à partir de laquelle le consignateur lit les messages de commande qui contrôlent son comportement.	SYSTEME SYSTEM.FTE.LOG.CMD. <i>nom_journal</i>
<code>wmqfte.queue.manager</code>	Gestionnaire de files d'attente auquel le consignateur se connecte. Ce paramètre est obligatoire et correspond à tout ce qui est nécessaire pour les connexions en mode liaison au gestionnaire de files d'attente. (Pour les propriétés de connexion à un gestionnaire de files d'attente éloignées, voir Tableau 112 , à la page 222.)	Aucune valeur par défaut
<code>wmqfte.message.source.type</code>	<p>L'une de valeurs suivantes :</p> <p>abonnement automatique Valeur par défaut. Le consignateur crée et utilise son propre abonnement durable et géré sur le gestionnaire de files d'attente défini dans SYSTEM.FTE/Log/#. Il s'agit d'une valeur appropriée pour la plupart des scénarios.</p> <p>abonnement administratif Si l'abonnement automatique n'est pas approprié, vous pouvez définir un autre abonnement (par exemple, à l'aide de IBM MQ Explorer, MQSC ou PCF) et demander au consignateur d'utiliser cet abonnement. Par exemple, utilisez cette valeur pour partitionner l'espace journal de sorte qu'un consignateur gère les agents de A-H, un autre consignateur gère I-P et un troisième consignateur de Q-Z.</p> <p>file d'attente Si la topologie IBM MQ signifie que la création d'un abonnement pour le consignateur n'est pas pratique, vous pouvez utiliser une file d'attente à la place. Configurez IBM MQ de sorte que la file d'attente reçoive les messages qui sont généralement reçus par un abonnement à SYSTEM.FTE/Log/# sur le gestionnaire de file d'attente de coordination.</p>	abonnement automatique
<code>wmqfte.message.source.name</code>	Si le type de source de messages est abonnement d'administration ou file d'attente, nom de l'abonnement ou de la file d'attente à utiliser. Cette propriété est ignorée si le type de source est abonnement automatique.	Aucune valeur par défaut

Tableau 111. Propriétés de connexion en mode liaison pour le fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<p><code>wmqfte.database.credentials.file</code></p>	<p>Fichier contenant le nom d'utilisateur et le mot de passe pour la connexion à la base de données.</p> <p>La valeur de cette propriété peut contenir des variables d'environnement.</p> <p>Pour plus d'informations, voir Format de fichier des données d'identification MFT.</p>	<p>z/OS Pour plus d'informations sur la création du fichier de données d'authentification, voir Configuration de MQMFTCredentials.xml sur z/OS.</p> <p>ALW Pour plus d'informations sur l'emplacement et les droits d'accès de ce fichier, voir Configuration de MQMFTCredentials.xml.</p> <p>ALW Voir aussi MFT et IBM MQ authentification de connexion.</p>
<p><code>wmqfte.database.driver</code></p>	<p>Emplacement des classes du pilote JDBC pour la base de données. Il s'agit généralement du chemin et du nom d'un fichier JAR.</p> <p>AIX Par exemple, le pilote de type 2 pour Db2 sur les systèmes AIX requiert le fichier <code>/opt/IBM/db2/V9.5/java/db2jcc.jar</code>.</p> <p>Windows Sur les systèmes Windows, spécifiez le séparateur de chemin sous la forme d'une barre oblique (<code>/</code>), par exemple, <code>C:/Program Files/IBM/SQLLIB/java/db2jcc.jar</code>.</p> <p>z/OS Sous z/OS, indiquez le chemin d'accès complet du fichier <code>db2jcc.jar</code>. Par exemple, <code>wmqfte.database.driver=/db2/db2v10/jdbc/classes/db2jcc.jar</code>.</p> <p>z/OS Sur les systèmes z/OS, vous devez référencer tous les fichiers JAR suivants:</p> <ul style="list-style-type: none"> • <code>db2jcc.jar</code> • <code>db2jcc_license_cisuz.jar</code> • <code>db2jcc_javax.jar</code> <p>Si votre pilote de base de données se compose de plusieurs fichiers JAR (par exemple, Db2 V9.1 requiert un fichier JAR de pilote et un fichier JAR de licence), incluez tous ces fichiers JAR dans cette propriété. Séparez plusieurs noms de fichier à l'aide du séparateur de chemin d'accès aux classes de votre plateforme, c'est-à-dire le point-virgule (<code>;</code>) sur les systèmes Windows et le signe deux-points (<code>:</code>) sur les autres plateformes.</p>	<p>Aucune valeur par défaut</p>

Tableau 111. Propriétés de connexion en mode liaison pour le fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.database.exclude_metadata</code>	<p>Contrôle si les entrées sont stockées dans la table de métadonnées qui contient des informations pouvant être trouvées dans d'autres tables du schéma du consignateur de base de données. Définissez cette valeur sur <code>true</code> ou sur <code>false</code>. Ces entrées de métadonnées ne sont plus stockées par défaut car il s'agit d'une duplication des données existantes et d'un gaspillage de la capacité de stockage de la base de données. Les entrées de propriété et les tables dans lesquelles les mêmes données apparaissent sont les suivantes:</p> <ul style="list-style-type: none"> • <code>com.ibm.wmqfte.SourceAgent</code> TRANSFER_EVENT ou CALL_REQUEST • <code>com.ibm.wmqfte.DestinationAgent</code> TRANSFER_EVENT • <code>com.ibm.wmqfte.MqmdUser</code> TRANSFER_EVENT ou CALL_REQUEST • <code>com.ibm.wmqfte.OriginatingUser</code> TRANSFER_EVENT ou CALL_REQUEST • <code>com.ibm.wmqfte.OriginatingHost</code> TRANSFER_EVENT ou CALL_REQUEST • <code>com.ibm.wmqfte.TransferId</code> TRANSFER ou CALL_REQUEST • <code>com.ibm.wmqfte.JobName</code> TRANSFER ou CALL_REQUEST <p>La définition de la valeur de cette propriété sur <code>false</code> entraîne le stockage de ces entrées de métadonnées dans la table de métadonnées.</p>	Oui
<code>wmqfte.database.host</code>	<p>Db2 uniquement :</p> <p>Nom d'hôte du serveur de base de données auquel se connecter à l'aide d'un pilote JDBC de type 4. Si une valeur est spécifiée pour cette propriété, une valeur pour <code>wmqfte.database.port</code> doit également être spécifiée. Si les deux propriétés ne sont pas définies, le consignateur de base de données se connecte à l'aide du pilote JDBC de type 2 par défaut.</p> <p>Si une valeur est spécifiée pour cette propriété, un fichier de données d'identification pour ce consignateur (chemin d'accès au fichier défini par la propriété <code>wmqfte.database.credentials.file</code>) doit exister et être accessible pour définir le nom d'utilisateur et le mot de passe de connexion à la base de données, même si la base de données se trouve sur le système local.</p>	Aucune valeur par défaut
<code>wmqfte.database.name</code>	Nom de l'instance de base de données (ou du sous-système lors de l'utilisation de Db2 for z/OS) qui contient les tables de journal Managed File Transfer .	Aucune valeur par défaut
<code>wmqfte.database.type</code>	Système de gestion de base de données en cours d'utilisation: Db2 ou Oracle. Définissez cette valeur sur <code>db2</code> ou <code>oracle</code> .	<code>db2</code>

Tableau 111. Propriétés de connexion en mode liaison pour le fichier `logger.properties` (suite)


Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.database.port</code>	<p>Db2 uniquement :</p> <p>Numéro de port du serveur de base de données auquel se connecter à l'aide d'un pilote JDBC de type 4. Si une valeur est spécifiée pour cette propriété, une valeur pour <code>wmqfte.database.host</code> doit également être spécifiée. Si les deux propriétés ne sont pas définies, le consigneur de base de données se connecte à l'aide du pilote JDBC de type 2 par défaut.</p> <p>Si une valeur est spécifiée pour cette propriété, un fichier de données d'identification pour ce consigneur (chemin d'accès au fichier défini par la propriété <code>wmqfte.database.credentials.file</code>) doit exister et être accessible pour définir le nom d'utilisateur et le mot de passe de connexion à la base de données, même si la base de données se trouve sur le système local.</p>	Aucune valeur par défaut
<code>wmqfte.database.schema</code>	<p>Db2 uniquement :</p> <p>Schéma de base de données qui contient les tables de journalisation Managed File Transfer. Dans la plupart des cas, la valeur par défaut est appropriée, mais vous devrez peut-être spécifier une autre valeur en fonction de vos propres considérations de base de données spécifiques au site.</p>	TELOG
<code>wmqfte.database.native.library.path</code>	<p>Chemin contenant les bibliothèques natives requises par le pilote de base de données de votre choix (le cas échéant).</p> <p> Par exemple, le pilote de type 2 pour Db2 sur les systèmes AIX requiert des bibliothèques de <code>/opt/IBM/db2/V9.5/lib32/</code>. Comme alternative à cette propriété, vous pouvez définir la propriété système <code>java.library.path</code> à l'aide d'autres méthodes.</p>	Aucune valeur par défaut
<code>wmqfte.file.logger.fileDirectory</code>	Répertoire dans lequel se trouvent les fichiers journaux du consigneur de fichiers.	<code>mqft/logs/coordination_dir/loggers/logger_name/logs</code>
<code>wmqfte.file.logger.fileSize</code>	Taille maximale qu'un fichier journal peut atteindre. La valeur de taille est un entier positif, supérieur à zéro, suivi de l'une des unités suivantes: Ko, Mo, Go, m (minutes), h (heures), d (jours), w (semaines). Par exemple, <code>wmqfte.file.logger.fileSize=5MB</code> indique une taille de fichier maximale de 5MB et <code>wmqfte.file.logger.fileSize=2d</code> indique une taille de fichier maximale de 2 jours de données.	10MB
<code>wmqfte.file.logger.fileCount</code>	Nombre maximal de fichiers journaux à créer. Lorsque la quantité de données dépasse la quantité maximale pouvant être stockée dans ce nombre de fichiers, le fichier le plus ancien est supprimé de sorte que le nombre de fichiers ne dépasse jamais la valeur spécifiée.	3

Tableau 111. Propriétés de connexion en mode liaison pour le fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<p><code>wmqfte.file.logger.mode</code></p>	<p>Mode de consignateur utilisé: circulaire ou linéaire. Définissez cette valeur sur <code>CIRCULAIRE</code> ou sur <code>LINEAR</code>.</p> <p>CIRCULAIRE-Le consignateur de fichiers écrit des informations dans un fichier jusqu'à ce que ce dernier atteigne sa taille maximale définie à l'aide de la propriété <code>wmqfte.file.logger.fileSize</code>. Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Le nombre maximal de fichiers écrits dans ce mode est contrôlé par la valeur définie à l'aide de la propriété <code>wmqfte.file.logger.fileCount</code>. Lorsque ce nombre maximal de fichiers est atteint, le consignateur de fichiers supprime le premier fichier et le recrée pour l'utiliser comme fichier actif. Si la valeur définie dans la propriété <code>wmqfte.file.logger.fileSize</code> est une unité d'octet de taille fixe (par exemple, Ko, Mo ou Go), la limite supérieure de l'espace disque utilisé dans ce mode est égale à <code>fileSize</code> multipliée par <code>fileCount</code>. Si la valeur définie dans la propriété <code>wmqfte.file.logger.fileSize</code> est une unité de temps (par exemple, m, h, dou w), la taille maximale dépend du débit des messages de journal dans votre système au cours de ces périodes. La convention de dénomination des fichiers journaux utilisée lors de l'exécution dans ce mode est: <code>logger_namenumber-timestamp.log</code> où:</p> <ul style="list-style-type: none"> • <code>logger_name</code> est le nom donné au consignateur dans la commande <code>fteCreateLogger</code>. • <code>nombre</code> est le numéro du fichier dans l'ensemble. • <code>horodatage</code> est l'horodatage de la création du fichier. <p>Exemple: <code>LOGGER1-20111216123430147.log</code></p> <p>LINEAR-Le consignateur de fichiers écrit des informations dans un fichier jusqu'à ce que ce fichier atteigne sa taille maximale définie à l'aide de la propriété <code>wmqfte.file.logger.fileSize</code>. Lorsque la taille maximale est atteinte, le consignateur de fichiers démarre un nouveau fichier. Les fichiers précédemment écrits ne sont pas supprimés, ce qui permet de les conserver en tant qu'enregistrement historique des messages de journal. Les fichiers ne sont pas supprimés lors de l'exécution en mode <code>linéaire</code>. Par conséquent, la propriété <code>wmqfte.file.logger.fileCount</code> est ignorée car il n'y a pas de limite supérieure au nombre de fichiers pouvant être créés. Etant donné qu'il n'y a pas de limite supérieure lors de l'exécution dans ce mode, il est nécessaire de suivre la quantité d'espace disque utilisée par les fichiers journaux afin d'éviter une faible quantité d'espace disque. La convention de dénomination des fichiers journaux utilisée lors de l'exécution dans ce mode est: <code>logger_name-timestamp.log</code> où:</p> <ul style="list-style-type: none"> • <code>logger_name</code> est le nom donné au consignateur dans la commande <code>fteCreateLogger</code>. • <code>horodatage</code> est l'horodatage de la création du fichier. <p>Exemple: <code>LOGGER-20111216123430147.log</code></p>	<p>Aucune valeur par défaut</p>

Tableau 111. Propriétés de connexion en mode liaison pour le fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.max.retry.interval</code>	<p>Durée maximale, en secondes, entre les nouvelles tentatives lorsque le consignateur rencontre une erreur persistante.</p> <p>Certaines conditions d'erreur (par exemple, la perte de la connexion à la base de données) empêchent le consignateur de continuer. Lorsque ce type de condition se produit, le consignateur annule la transaction en cours, attend un certain temps, puis effectue de nouvelles tentatives. Le temps d'attente du consignateur est initialement très court, de sorte que les erreurs transitoires peuvent être surmontées rapidement. Toutefois, à chaque nouvelle tentative du consignateur, le temps d'attente est augmenté. Cela évite trop de travail inutile lorsque la condition d'erreur est plus longue, par exemple lorsqu'une base de données est arrêtée à des fins de maintenance.</p> <p>Utilisez cette propriété pour définir une limite à la durée de l'attente de sorte qu'une nouvelle tentative se produise dans un délai raisonnable après la résolution de la condition d'erreur.</p>	600
<code>immediateShutdownTimeout</code>	<p>Durée, en secondes, pendant laquelle le consignateur attend que toutes les opérations en attente se terminent et s'arrêtent correctement. Par défaut, le consignateur attend pendant 10 secondes que les opérations se terminent. Si les opérations ne sont pas terminées avant le délai d'attente, le consignateur écrit le message d'événement suivant dans <code>output0.log</code> et se termine.</p> <p><code>BFGDB0082I: The logger is ending immediately.</code></p> <p>Si vous indiquez la valeur zéro, le consignateur attend que les opérations en cours soient terminées.</p> <p>La valeur par défaut est utilisée si la valeur de <code>immediateShutdownTimeout</code> est inférieure à zéro.</p> <p>La propriété s'applique à la fois au consignateur de base de données autonome et au consignateur de type de fichier.</p>	10
<code>loggerCredentialsKeyFile</code>	Nom du fichier contenant la clé de données d'identification utilisée lors du chiffrement des données d'identification.	Propriété de chaîne sans valeur par défaut.
<code>loggerQMGrRetryInterval</code>	Intervalle, en secondes, entre les vérifications de la disponibilité du gestionnaire de files d'attente par le contrôleur de processus du consignateur.	30
<code>maxRestartCount</code>	Nombre maximal de redémarrages pouvant se produire dans l'intervalle de temps spécifié par la valeur de la propriété d'intervalle <code>maxRestart</code> . Lorsque cette valeur est dépassée, le contrôleur de processus du consignateur arrête de redémarrer le consignateur et effectue à la place une action basée sur la valeur de la propriété de délai <code>maxRestart</code> .	4
<code>maxRestartInterval</code>	Intervalle, en secondes, pendant lequel le contrôleur de processus du consignateur mesure les redémarrages du consignateur. Si le nombre de redémarrages dans cet intervalle dépasse la valeur de la propriété Nombre <code>maxRestart</code> , le contrôleur de processus du consignateur arrête de redémarrer le consignateur. A la place, le contrôleur de processus du consignateur effectue une action basée sur la valeur de la propriété de délai <code>maxRestart</code> .	120

Tableau 111. Propriétés de connexion en mode liaison pour le fichier logger.properties (suite)




Nom de la propriété	Description	Valeur par défaut
maxRestartDelay	Détermine le comportement du contrôleur de processus du consignateur lorsque le taux de redémarrages du consignateur dépasse la valeur des propriétés d'intervalle maxRestartet maxRestart. Si vous indiquez une valeur inférieure ou égale à zéro, le contrôleur de processus du consignateur est arrêté. Si vous indiquez une valeur supérieure à zéro, il s'agit du nombre de secondes à attendre avant que les informations d'historique de redémarrage détenues par le contrôleur de processus du consignateur soient réinitialisées et que le consignateur soit redémarré.	-1
wmqfte.oracle.port	Port utilisé par le consignateur pour se connecter à l'instance Oracle . Ce port est également appelé programme d'écoute TNS.	1521
wmqfte.oracle.host	Hôte utilisé par le consignateur pour se connecter à l'instance Oracle .	système hôte local
armELEMTYPE	Propriété facultative. Si le consignateur est configuré pour être redémarré par le gestionnaire de redémarrage automatique (ARM), définissez cette propriété sur la valeur du paramètre ARM ELEMTYPE spécifiée dans la règle ARM associée. Pour un consignateur, définissez ELEMTYPE sur SYSBFGLG.	Non définie
armELEMENT	Propriété facultative. Si le consignateur est configuré pour être redémarré par le gestionnaire de redémarrage automatique (ARM), définissez cette propriété sur la valeur du paramètre ARM ELEMENT spécifiée dans la règle ARM associée. Vous pouvez définir la valeur ELEMENT pour qu'elle corresponde au nom du consignateur.	Non définie
loggerQMGrAuthenticationCredentials File	Chemin d'accès au fichier qui contient les données d'identification de connexion MQ pour la connexion au gestionnaire de files d'attente de coordination du consignateur.	<p> Pour plus d'informations sur la création du fichier de données d'authentification, voir Configuration de MQMFTCredentials.xml sur z/OS.</p> <p> Pour plus d'informations sur l'emplacement et les droits d'accès à ce fichier, voir Configuration de MQMFTCredentials.xml.</p> <p> Voir aussi MFT et IBM MQ authentification de connexion.</p>
trace	Propriété facultative. Spécification de trace lorsque le consignateur doit être exécuté avec la trace activée au démarrage du consignateur. La spécification de trace est une liste de classes séparée par des virgules, suivie du signe égal et d'un niveau de trace. Par exemple, com.ibm.wmqfte.databaseloggeret com.ibm.wmqfte.databaselogger.operation =all Vous pouvez spécifier plusieurs spécifications de trace dans une liste de valeurs séparées par deux-points. Exemple: com.ibm.wmqfte.databaselogger=moderate: com.ibm.wmqfte.databaselogger.operation =all	Aucun

Tableau 111. Propriétés de connexion en mode liaison pour le fichier <i>logger.properties</i> (suite)		
Nom de la propriété	Description	Valeur par défaut
traceFiles	Propriété facultative. Nombre total de fichiers de trace à conserver. Cette valeur s'applique au contrôleur de processus d'un consignateur, ainsi qu'au consignateur lui-même.	5
traceSize	Propriété facultative. Taille maximale en Mo de chaque fichier de trace, avant que la trace passe au fichier suivant. Cette valeur s'applique au contrôleur de processus du consignateur et au consignateur lui-même.	20
wmqfte.file.logger.filePermissions	<p>Propriété facultative. Permet d'indiquer le type de droit requis pour le fichier journal du consignateur.</p> <p>La propriété s'applique aux journaux linéaires et circulaires et peut prendre les valeurs <i>UserReadWriteOnly</i> ou <i>UserReadWriteAllRead</i>.</p> <p>La valeur <i>UserReadWriteOnly</i> a les droits équivalents existants de 600 et la valeur <i>UserReadWriteAllRead</i> a les droits équivalents de 644.</p> <p>Toute modification des droits d'accès est applicable aux fichiers du consignateur nouvellement créés.</p> <p>Si vous entrez une valeur pour la propriété qui n'est pas valide, le consignateur prend la valeur par défaut et émet le message BFGDB0083W dans le journal de sortie.</p>	UserReadWriteOnly

Propriétés de connexion SSL/TLS en mode client

Propriétés requises pour la prise en charge de la connexion en mode client à un gestionnaire de files d'attente de consignateur à l'aide de SSL/TLS.

Tableau 112. Propriétés de connexion SSL/TLS en mode client pour le fichier <i>logger.properties</i>		
Nom de la propriété	Description	Valeur par défaut
wmqfte.queue.manager.host	Nom d'hôte ou adresse IP du gestionnaire de files d'attente du consignateur.	Aucune valeur par défaut
wmqfte.queue.manager.port	Port sur lequel le gestionnaire de files d'attente du consignateur est en mode écoute.	1414
wmqfte.queue.manager.channel	Nom du canal de connexion serveur sur le gestionnaire de files d'attente du consignateur.	SYSTEM.DEF.SVRCONN
wmqfte.Ssl.CipherSuite	<p>Indique les aspects TLS de la façon dont le consignateur et le gestionnaire de files d'attente du consignateur échangent des données.</p> <p>La valeur de wmqfte.Ssl.CipherSuite est un nom CipherSuite . Le nom CipherSuite est mappé au nom CipherSpec utilisé sur le canal du gestionnaire de files d'attente du consignateur.</p> <p>Pour plus d'informations, voir CipherSuite et CipherSpec name mappings.</p>	Aucune valeur par défaut
wmqfte.Ssl.PeerName	Indique un squelette de nom distinctif qui doit correspondre au nom fourni par le gestionnaire de files d'attente du consignateur. Le nom distinctif est utilisé pour vérifier le certificat d'identification présenté par le gestionnaire de files d'attente lors de la connexion.	Aucune valeur par défaut

Tableau 112. Propriétés de connexion SSL/TLS en mode client pour le fichier `logger.properties` (suite)

Nom de la propriété	Description	Valeur par défaut
<code>wmqfte.Ssl.TrustStore</code>	<p>Indique l'emplacement des certificats auxquels le consignateur fait confiance. La valeur de wmqfte.Ssl.TrustStore est un chemin de fichier.</p> <p>Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappées à l'aide d'une barre oblique inversée (\). Windows</p> <p>Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/).</p> <p>Notez que la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucune valeur par défaut
<code>wmqfte.Ssl.TrustStoreCredentialsFile</code>	<p>Chemin d'accès au fichier qui contient les données d'identification wmqfte.Ssl.TrustStore.</p> <p>Notez que la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucune valeur par défaut
<code>wmqfte.Ssl.TrustStoreType</code>	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>
<code>wmqfte.Ssl.KeyStore</code>	<p>Indique l'emplacement de la clé privée du consignateur. La valeur de wmqfte.Ssl.KeyStore est un chemin de fichier.</p> <p>Les parenthèses, les virgules (,) et les barres obliques inversées (\) sont des caractères spéciaux dans les commandes MFT et doivent être échappées à l'aide d'une barre oblique inversée (\). Windows</p> <p>Les chemins d'accès aux fichiers sous Windows peuvent être spécifiés à l'aide de doubles barres obliques inversées (\\) comme séparateur ou à l'aide de barres obliques simples (/).</p> <p>Notez que la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucune valeur par défaut
<code>wmqfte.Ssl.KeyStore.CredentialsFile</code>	<p>Chemin d'accès au fichier qui contient les données d'identification wmqfte.Ssl.KeyStore.</p> <p>Notez que la valeur de cette propriété peut contenir des variables d'environnement.</p>	Aucune valeur par défaut
<code>wmqfte.Ssl.KeyStoreType</code>	Type de magasin de clés SSL que vous souhaitez utiliser. Les magasins de clés JKS et PKCS#12 sont pris en charge. La valeur de cette propriété peut être <code>jks</code> ou <code>pkcs12</code> .	<code>jks</code>
<code>wmqfte.Ssl.FipsRequired</code>	Indique que vous souhaitez activer la prise en charge de FIPS au niveau du consignateur. La valeur de cette propriété peut être <code>true</code> ou <code>false</code> . Pour plus d'informations, voir Prise en charge de FIPS dans MFT .	<code>false</code>

Concepts associés

Propriétés SSL/TLS pour MFT

Référence associée

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 175

Il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction de l'environnement en cours (par exemple, l'utilisateur exécutant une commande).

«Le fichier MFT `agent.properties`», à la page 181

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, `agent.properties`, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier `agent.properties` peut également contenir des propriétés qui modifient le comportement de l'agent.

«Le fichier MFT `command.properties`», à la page 210

Le fichier `command.properties` indique le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente.

«Le fichier MFT `coordination.properties`», à la page 205

Le fichier `coordination.properties` indique les détails de connexion au gestionnaire de file d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier `coordination.properties` commun sur une unité partagée.

Sortie générée par la fonction LogTransfer

Les événements du journal de transfert capturent les détails de la progression du transfert à partir du moment où le transfert est soumis jusqu'à sa fin. Les informations sur le transfert en cours de resynchronisation sont également capturées pour vous aider à comprendre la progression d'un transfert.

Format d'événement de transfert

Les événements de transfert sont au format JSON et écrits dans le fichier `transferlogN.json`, qui est créé dans le répertoire de journaux de l'agent, où N est un nombre avec 0 comme valeur par défaut. Chaque événement inclut les attributs communs suivants:

- Date et heure (en temps universel coordonné)
- Identifiant unique

Des attributs supplémentaires sont inclus dans les informations d'événement écrites, en fonction du type de l'événement et du niveau du journal de transfert. Alors que le niveau de journalisation de transfert *info* écrit des informations minimales, le niveau *verbose* inclut des informations plus détaillées. La section «Exemples d'événements», à la page 225 suivante décrit quelques exemples d'événements de transfert consignés par un agent.

Identifiant unique

L'ID unique est inclus pour vous aider à identifier facilement les différentes phases au fur et à mesure de la progression d'un transfert, par exemple, BFGTL0001. L'ID unique fait partie de l'attribut **eventDescription** et se compose de deux parties:

BFGTL

Préfixe utilisé pour tous les identificateurs, où BFG est le suffixe standard utilisé dans Managed File Transfer et TL indique qu'il s'agit d'un journal de transfert.

Numéro

Numéro unique commençant par 1. Exemple :

```
{
  "eventDescription": "BFGTL0001: New transfer request submitted"
}
```

Exemples d'événements

Le tableau suivant décrit certains des événements en tant qu'exemples des informations consignées par la fonction supplémentaire. La deuxième colonne de la table *Niveau de journalisation* indique le niveau auquel l'événement est consigné.

Important : Les attributs suivants sont inclus dans les informations d'événement si le niveau **logTransfer** est défini sur *verbose* ou *modéré*:

- **sourceAgent**
- **destinationAgent**
- **threadId**

Événement	Niveau de journalisation	Description
Liste des éléments à transférer	prolix	<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"}] }</pre> <p>Example:</p> <pre>{ "dateTime": "2022-01-14T12:56:54.219Z UTC", "eventDescription": "BFGTL0002I: Generated detailed transfer item list.", "destinationAgent": "QMBAGQ", "sourceAgent": "QMBAG1", "threadId": "0000001d", "totalItems": 1, "transferId": "414d5120514d41202020202020202063bd17610a390040", "transferItems": [{" "destination": "/results/rts/target/destFile.txt", "source": "DESTINATIONQQ@QMB" }] }</pre>

Événement	Niveau de journalisation	Description
Liste des transferts à récupérer au démarrage de l'agent	prolix	<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>"}] "threadId": "<Thread Id>", }</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": "CQMHX01AG1", "transfers": [{"414D512043514D485830312020202020B0D4176101370040": "completeReceived" }, {"414D512043514D485830312020202020B0D4176101370050": "resynchronizing"}] "threadId": "0000001c", }</pre>

Référence associée

«Propriétés système Java pour MFT», à la page 227

Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java , car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

[fteCreateAgent](#)

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 175

Il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction de l'environnement en cours (par exemple, l'utilisateur exécutant une commande).

Propriétés système Java pour MFT




Un certain nombre de propriétés de commande et d'agent Managed File Transfer doivent être définies en tant que propriétés système Java , car elles définissent la configuration de la fonction précoce qui ne peut pas utiliser le mécanisme des propriétés de commande ou d'agent.

Définissez les propriétés système et les autres options JVM pour la machine virtuelle Java qui doit exécuter les commandes Managed File Transfer en définissant la variable d'environnement BFG_JVM_PROPERTIES. Par exemple, pour définir la propriété com.ibm.wmqfte.maxConsoleLineLength sur une plateforme de type UNIX, définissez la variable comme suit:

```
export BFG_JVM_PROPERTIES="-Dcom.ibm.wmqfte.maxConsoleLineLength=132"
```

Si vous exécutez un agent en tant que service Windows , vous pouvez modifier les propriétés système Java de l'agent en spécifiant le paramètre `-sj` dans la commande **fteModifyAgent** .

Tableau 113. JavaPropriétés système

Nom de la propriété	Description	Valeur
com.ibm.wmqfte.maxConsoleLineLength	Longueur maximale de la ligne pouvant être écrite sur la console. Les lignes qui dépassent cette longueur sont des retours à la ligne. Cette valeur est exprimée en octets (et non en caractères).	 La longueur par défaut de l' IBM i est de 132 octets.  Pour AIX, Linux, and Windows, la longueur est illimitée.  Pour z/OS, la longueur est illimitée.
com.ibm.wmqfte.daemon.windows.windowsServiceLogFilesm	(Windows uniquement.) Indique le nombre maximal de fichiers journaux de service Windows à conserver. Les fichiers journaux du service Windows sont créés dans les répertoires des journaux de l'agent et du consignateur de base de données si ces applications s'exécutent en tant que service Windows . Les fichiers journaux du service Windows sont nommés avec le préfixe <i>service</i> et contiennent des messages sur le démarrage et l'arrêt du service.	5

Concepts associés

[Options de configuration de MFT sur Multiplatforms](#)

[Conseils et astuces pour l'utilisation d' MFT](#)

CipherSpecs et suites de chiffrement SHA-2 pour MFT

Managed File Transfer prend en charge les CipherSpecs et CipherSuites SHA-2 .

Pour plus d'informations sur les CipherSpecs et CipherSuites disponibles pour les connexions entre les agents et les gestionnaires de files d'attente IBM MQ , voir [TLS CipherSpecs et CipherSuites dans IBM MQ classes for Java](#) et [SSL/TLS CipherSpecs et CipherSuites dans IBM MQ classes for JMS](#).

Pour plus d'informations sur la configuration de CipherSpecs et de CipherSuites à utiliser avec les agents de pont de protocole (PBA) et les serveurs FTPS, voir [Prise en charge du serveur FTPS par le pont de protocole](#) et [Format de fichier de propriétés du pont de protocole](#).

Si vous souhaitez vous conformer à la norme SP 800-131A, vous devez respecter les exigences suivantes:

- Vous devez utiliser FTPS, que vous avez configuré de manière appropriée ; SFTP n'est pas pris en charge.
- Le serveur distant doit envoyer des suites de chiffrement SP 800-131A-compliant uniquement.

Concepts associés

[Propriétés SSL/TLS pour MFT](#)

Fichiers de configuration du consignateur de fichiers MFT

Outre le fichier `logger.properties` , un consignateur de fichier autonome Managed File Transfer dispose également d'un fichier de configuration XML dans son répertoire de configuration. Ce fichier de configuration est appelé `FileLoggerFormat.xml` et il définit le format utilisé par le consignateur de fichiers pour écrire des messages dans le fichier journal. Le contenu de ce fichier doit être conforme au schéma XML défini dans le fichier `FileLoggerFormat.xsd` .

Concepts associés

[Format du consignateur de fichiers MFT autonome](#)

Référence associée

«Le fichier MFT logger.properties», à la page 214

Le consignateur Managed File Transfer possède un ensemble de propriétés de configuration. Spécifiez ces propriétés dans le fichier `logger.properties`, qui se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`.

«Format de journal par défaut du consignateur de fichiers autonome MFT», à la page 229

Définition du format de fichier journal par défaut pour le consignateur de fichier Managed File Transfer autonome.

«Format XSD du consignateur de fichier autonome», à la page 234

Schéma d'un format de fichier autonome.

Format de journal par défaut du consignateur de fichiers autonome MFT

Définition du format de fichier journal par défaut pour le consignateur de fichier Managed File Transfer autonome.

```
<?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>
  </messageTypes>
</logFormatDefinition>
```

```

</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/action</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>
            <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 |

```



```

</format>
</transferComplete>
<transferDelete>
  <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/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>

```

Référence associée

[Format du consignateur de fichiers MFT autonome](#)

«Format XSD du consignateur de fichier autonome», à la page 234

Schéma d'un format de fichier autonome.

Format XSD du consignateur de fichier autonome

Schéma d'un format de fichier autonome.

Schéma

```
<?xml version="1.0" encoding="UTF-8"?>
<!--
@start_non_restricted_prolog@
Version: %Z% %I% %W% %E% %U% [%H% %T%]

Licensed Materials - Property of IBM

5724-H72

Copyright IBM Corp. 2011, 2024. All Rights Reserved.

US Government Users Restricted Rights - Use, duplication or
disclosure restricted by GSA ADP Schedule Contract with
IBM Corp.
@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:sequence>
</xsd:complexType>

```

```

        <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:restriction>
  </xsd:simpleType>
</xsd:schema>

```

Référence associée

[Format du consignateur de fichiers MFT autonome](#)

«Format de journal par défaut du consignateur de fichiers autonome MFT», à la page 229

Définition du format de fichier journal par défaut pour le consignateur de fichier Managed File Transfer autonome.

z/OS Bibliothèque SCSQFCMD

La bibliothèque SCSQFCMD fournie par IBM MQ Managed File Transfer for z/OS contient des membres qui agissent en tant que modèles pour les travaux qui peuvent être utilisés pour créer une configuration Managed File Transfer , ainsi que pour créer et administrer un agent ou un consignateur.

Le contenu de la bibliothèque est présenté dans le tableau suivant.

Membre	Description
BFGCOPY	Travail utilisé pour créer une copie de la bibliothèque SCSQFCMD
BFGCUSTM	Travail utilisé pour personnaliser une copie de la bibliothèque pour un agent ou un consignateur
BFGXCROB	Exemple de modèle fteObfuscate .
BFGXLGCR	Modèle fteCreateLogger .
BFGXMNCR	Exemple de modèle fteCreateMonitor .
BFGXMNDE	Exemple de modèle fteDeleteMonitor .
BFGXPRAN	fteAnt Exemple de modèle
BFGXSTDE	fteDeleteScheduledTransfer Exemple de modèle
BFGXTMCR	fteCreateTemplate Exemple de modèle
BFGXTMDE	fteDeleteTemplate Exemple de modèle
BFGXTRCA	fteCancelTransfer Exemple de modèle
BFGXTRCR	fteCreateTransfer Exemple de modèle
BFGYAGST	Modèle de procédure de tâche démarrée pour démarrer un agent
BFGYLGST	Modèle de procédure de tâche démarrée pour démarrer un consignateur
BFGZAGCL	fteCleanAgent Exemple de modèle
BFGZAGCR	fteCreateAgent Exemple de modèle
BFGZAGDE	fteDeleteAgent Exemple de modèle
BFGZAGLG	fteSetAgentLogLevel Exemple de modèle
BFGZAGLI	fteListAgents Exemple de modèle
BFGZAGPI	ftePingAgent Exemple de modèle
BFGZAGSH	fteShowAgentDetails Exemple de modèle
BFGZAGSP	fteStopAgent Exemple de modèle
BFGZAGST	fteStartAgent Exemple de modèle
BFGZAGTC	fteSetAgentTraceLevel Exemple de modèle

Membre	Description
BFGZCFRC	fteSetupCoordination Exemple de modèle
BFGZCFDF	fteChangeDefaultConfigurationOptions Exemple de modèle
BFGZCMCR	fteSetupCommands Exemple de modèle
Commande BFGZCMD	Modèle de script REXX utilisé par d'autres membres du jeu de données
BFGZLGDE	fteDeleteLogger Exemple de modèle
BFGZLGSH	fteShowLoggerDetails Exemple de modèle
BFGZLGSP	fteStopLogger Exemple de modèle
BFGZLGST	fteStartLogger Exemple de modèle
BFGZLGTC	fteSetLoggerTraceLevel Exemple de modèle
BFGZMCLI	fteListMonitors Exemple de modèle
ID de processus BFGZPID	fteSetProductId Exemple de modèle
BFGZPROF	Modèle de script shell utilisé par d'autres membres du fichier
Commande BFGZPRSH	fteDisplayVersion Exemple de modèle
BFGZRAS	fteRas Exemple de modèle
BFGZSTLI	fteListScheduledTransfers Exemple de modèle
BFGZTMLI	fteListTemplates Exemple de modèle

Pour plus de détails sur la façon dont la bibliothèque SCSQFCMD est utilisée pour générer une nouvelle bibliothèque pour la création d'une configuration Managed File Transfer et la création et l'administration d'un agent ou d'un consignateur, voir [Configuration de Managed File Transfer for z/OS](#).

Référence associée

«Utilisation des variables d'environnement dans les propriétés MFT», à la page 175

Il est possible d'utiliser des variables d'environnement dans les propriétés Managed File Transfer qui représentent des emplacements de fichier ou de répertoire. Cela permet aux emplacements des fichiers ou des répertoires utilisés lors de l'exécution de parties du produit de varier en fonction de l'environnement en cours (par exemple, l'utilisateur exécutant une commande).

Rubrique SYSTEM.FTE

SYSTEM.FTE est une rubrique du gestionnaire de file d'attente de coordination utilisée par Managed File Transfer pour consigner des transferts et stocker des informations sur les agents, les moniteurs, les plannings et les modèles.

Structure de la rubrique

```
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/nom_agent

Cette rubrique contient une publication conservée qui décrit un agent sur votre réseau Managed File Transfer, et ses propriétés. Le message dans cette rubrique est mis à jour régulièrement avec le statut de l'agent. Pour plus d'informations, voir [Format de message de statut d'agent MFT](#).

SYSTEM.FTE/monitors/nom_agent

Cette rubrique contient des publications conservées qui décrivent les moniteurs de ressources associés à l'agent *nom_agent*. Le fichier XML de la publication conservée est conforme au schéma `MonitorList.xsd`. Pour plus d'informations, voir [Format de message de liste de moniteurs MFT](#).

SYSTEM.FTE/Scheduler/nom_agent

Cette rubrique contient une publication conservée qui décrit toutes les planifications actives qui sont associées à l'agent *nom_agent*. Le fichier XML de la publication conservée est conforme au schéma `ScheduleList.xsd`. Pour plus d'informations, voir [Format de message de liste de planifications MFT](#).

SYSTEM.FTE/Templates

Cette rubrique contient des publications conservées qui décrivent tous les modèles qui sont définis dans votre topologie Managed File Transfer.

- La publication associée à chaque modèle est publiée dans une sous-rubrique nommée `SYSTEM.FTE/Templates/template_ID`.

Pour un exemple de contenu de cette publication conservée, voir [Exemple de message XML de modèle MFT](#).

SYSTEM.FTE/Transfers/nom_agent

Cette rubrique contient des publications qui décrivent le statut des transferts depuis l'agent *nom_agent*. Les publications associées à chaque transfert sont publiées dans une sous-rubrique portant le nom `SYSTEM.FTE/Transfers/agent_name/transfer_ID`. Ces publications sont utilisées par le plug-in IBM MQ Explorer pour fournir des informations sur les transferts individuels. Le fichier XML de la publication est conforme au schéma `TransferStatus.xsd`. Pour plus d'informations, voir [Format de message de statut de transfert de fichier](#).

SYSTEM.FTE/Log/nom_agent

Cette rubrique contient des publications qui consignent les informations sur les transferts, les moniteurs et les planifications provenant de l'agent *nom_agent*. Ces publications peuvent être consignées par le consignateur de base de données pour fournir des enregistrements d'audit des événements qui surviennent sur votre réseau Managed File Transfer.

- Les publications associées à chaque transfert sont publiées dans une sous-rubrique portant le nom `SYSTEM.FTE/Log/agent_name/transfer_ID` et le code XML de la publication est conforme au schéma `TransferLog.xsd`. Pour plus d'informations, voir [Formats de message de journal de transfert de fichier](#).
- Les publications associées à chaque transfert planifié sont publiées dans une sous-rubrique portant le nom `SYSTEM.FTE/Log/agent_name/schedule_ID` et le code XML de la publication est conforme au schéma `ScheduleLog.xsd`. Pour plus d'informations, voir [Formats de message de journal de transfert de fichier planifié](#).
- Les publications associées à chaque moniteur sont publiées dans une sous-rubrique nommée `SYSTEM.FTE/Log/agent_name/Monitors/monitor_name/monitor_ID` et le code XML de la publication est conforme au schéma `MonitorLog.xsd`. Pour plus d'informations, voir [Format de message de journal de moniteur MFT](#).

Paramètres de file d'attente de l'agent MFT

Les scripts de commande MQSC générés par la commande **fteCreateAgent** créent les files d'attente d'agent avec les paramètres définis sur les valeurs suivantes. Si vous n'utilisez pas les scripts MQSC fournis pour créer les files d'attente, mais que vous les créez manuellement, veuillez à définir les paramètres suivants sur les valeurs indiquées.

Files d'attente d'opérations d'agent

Les files d'attente d'opérations de l'agent portent les noms suivants:

- SYSTEME SYSTEM.FTE.COMMAND.*nom_agent*
- SYSTEME SYSTEM.FTE.DATA.*nom_agent*
- SYSTEME SYSTEM.FTE.EVENT.*nom_agent*
- SYSTEME SYSTEM.FTE.REPLY.*nom_agent*
- SYSTEME SYSTEM.FTE.STATE.*nom_agent*

Paramètre	Valeur (le cas échéant)
DEFPRTY	0
DEFSOPT	PARTAGÉ
GET	Activée
MAXDEPTH	5000
MAXMSGL	4194304
MSGDLVSQ	PRIORITY
PUT	Activée
RETINTVL	999999999
PARTAGER	
Pas de déclencheur	
SYNTAXE	NORMALE
REPLACE	

Files d'attente d'autorisations d'agent

Les files d'attente de droits de l'agent portent les noms suivants:

- SYSTEM.FTE.AUTHADM1.*nom_agent*
- SYSTEME SYSTEM.FTE.AUTHAGT1. *nom_agent*
- SYSTEM.FTE.AUTHMON1.*nom_agent*
- SYSTEM.FTE.AUTHOPS1.*nom_agent*
- SYSTEM.FTE.AUTHSCH1.*nom_agent*
- SYSTEM.FTE.AUTHTRN1.*nom_agent*

Paramètre	Valeur (le cas échéant)
DEFPRTY	0
DEFSOPT	PARTAGÉ

Paramètre	Valeur (le cas échéant)
GET	Activée
MAXDEPTH	0
MAXMSGL	0
MSGDLVSQ	PRIORITY
PUT	Activée
RETINTVL	999999999
PARTAGER	
Pas de déclencheur	
SYNTAXE	NORMALE
REPLACE	

Référence associée

Agent fteCreate(création d'un agent MFT)

Les files d'attente système MFT et la rubrique système

Managed File Transfer possède plusieurs files d'attente système et une rubrique système à usage interne uniquement.

Toutes les files d'attente dont le nom commence par SYSTEM.FTE sont des files d'attente système internes pour Managed File Transfer (MFT). Ne supprimez pas ces files d'attente car cela empêche IBM MQ MFT de fonctionner correctement. [Tableau 116](#), à la page 240 indique le type de message dans chaque file d'attente:

Nom de la file d'attente	Type de file d'attente	Utilisation
SYSTEM.FTE.AUTHAGT1.agent_name	Droits d'accès	File d'attente pour la configuration des droits d'envoi et de réception des demandes de transfert.
SYSTEM.FTE.AUTHTRN1.agent_name	Droits d'accès	File d'attente pour la configuration des droits de démarrage et d'annulation des transferts gérés. Permet également de démarrer des appels gérés.
SYSTEM.FTE.AUTHMON1.agent_name	Droits d'accès	File d'attente pour la configuration des droits permettant à un utilisateur de créer ou de supprimer des moniteurs de ressources créés par le même utilisateur.

Tableau 116. Noms de file d'attente, type et utilisation (suite)

Nom de la file d'attente	Type de file d'attente	Utilisation
SYSTEM.FTE.AUTHOPS1.agent_name	Droits d'accès	File d'attente pour la configuration des droits de suppression des moniteurs de ressources et des transferts planifiés créés par un autre utilisateur.
SYSTEM.FTE.AUTHSCH1.agent_name	Droits d'accès	File d'attente pour la configuration des droits permettant de créer ou de supprimer des transferts planifiés créés par le même utilisateur.
SYSTEM.FTE.AUTHADM1.agent_name	Droits d'accès	File d'attente pour la configuration des droits d'arrêt de l'agent, à l'aide de l'option -m de la commande fteStopAgent .
SYSTEM.FTE.COMMAND.agent_name	Opération	File d'attente pour l'envoi de demandes de commande à un agent.
SYSTEM.FTE.DATA.agent_name	Opération	File d'attente utilisée par un agent de destination pour le stockage des données envoyées par un agent source.
SYSTEM.FTE.REPLY.agent_name	Opération	File d'attente de réception des réponses d'un agent de destination.
SYSTEM.FTE.STATE.agent_name	Opération	File d'attente pour le maintien du statut d'une demande de transfert.
SYSTEM.FTE.EVENT.agent_name	Opération	File d'attente pour le maintien de l'historique du moniteur de ressources.
SYSTEM.FTE.HA.agent_name	Opération	File d'attente utilisée comme verrou par les instances d'agent à haute disponibilité.

Si un agent participe à des transferts de message à fichier ou de fichier à message, il se peut que la définition de la file d'attente SYSTEM.FTE.STATE.agent_name doive être modifiée pour permettre ces transferts gérés. Pour plus d'informations à ce sujet, voir [Conseils pour la définition des attributs MQ et des propriétés MFT associées à la taille des messages](#).



Avertissement : Vous ne devez pas modifier les définitions des autres files d'attente système.

De plus, ne modifiez pas et ne supprimez pas SYSTEM.FTE, car cette rubrique est également destinée à un usage interne uniquement.

Files d'attente temporaires

Managed File Transfer crée des files d'attente temporaires à différentes fins. Le nom de chaque file d'attente commence par WMQFTE. par défaut. (Le point fait partie du préfixe par défaut.)

Pour changer ce préfixe, vous pouvez utiliser la propriété **dynamicQueuePrefix** dans le fichier `command.properties` ou le fichier `coordination.properties`, ou dans les deux. La propriété dans le fichier `command.properties` permet de définir le préfixe des files d'attente temporaires qui sont créées pour les réponses aux commandes requérant une réponse de l'agent. La propriété dans le fichier `coordination.properties` permet de définir le préfixe des files d'attente temporaires qui sont créées à d'autres fins, comme `WMQFTE.FTE.TIMECHCK.QUEUE`, où `WMQFTE` est la valeur définie par la propriété **dynamicQueuePrefix**.

Référence associée


[Restriction des droits utilisateur sur les actions de l'agent MFT](#)

Conventions de dénomination des objets MFT

Utilisez les conventions de dénomination suivantes pour vos objets Managed File Transfer :

- Noms d'agent et de consignateur:
 - Peut comporter jusqu'à 28 caractères et n'est pas sensible à la casse.
 - Les caractères entrés en minuscules ou en casse mixte sont convertis en majuscules
 - Doit être conforme aux conventions de dénomination d'objet IBM MQ standard.Ces conventions sont détaillées comme suit: [Règles de dénomination des objets IBM MQ](#).
- En plus des conventions de dénomination des objets IBM MQ , les règles suivantes s'appliquent:
 - Le caractère barre oblique (/) ne peut pas être utilisé dans les noms d'agent ou de consignateur
 - Le caractère pourcentage (%) ne peut pas être utilisé dans les noms d'agent ou de consignateur.
- Les noms des propriétés dans les fichiers de propriétés sont sensibles à la casse.
- Les noms de gestionnaire de files d'attente sont sensibles à la casse.
- Les noms de fichier sont sensibles à la casse pour certaines plateformes.
- Noms de modèle de transfert et de moniteur de ressources:
 - Ne sont pas sensibles à la casse
 - Les caractères entrés en minuscules ou en casse mixte sont convertis en majuscules
 - Ne doit pas contenir d'astérisque (*), de pourcentage (%) ou de point d'interrogation (?)
- Les noms de serveur de fichiers de protocole doivent:
 - Être d'une longueur minimale de 2 caractères, mais il n'y a pas de limite de longueur maximale
 - Ne sont pas sensibles à la casse
 - Doit être conforme aux conventions de dénomination d'objet IBM MQ standard.Ces conventions sont détaillées comme suit: [Règles de dénomination des objets IBM MQ](#).


Fichiers du système de fichiers intégré IBM i (IFS)

 Les noms de fichier du système de fichiers intégré ne peuvent contenir aucun des caractères suivants:

- Barre oblique inversée (\)
- la barre oblique (/)
- Deux-points (:)
- Astérisque (*)
- Point d'interrogation (?)
- Guillemets (")
- Symbole inférieur à (<)
- Symbole supérieur à (>)
- Barre verticale (|)

Si vous tentez de transférer des fichiers dont le nom contient l'un de ces caractères vers un système IBM i IFS, le transfert de ces fichiers échoue.

Noms de fichier

 Les fichiers sont soumis à des restrictions de dénomination qui affectent la longueur maximale du nom et les caractères disponibles que vous pouvez utiliser pour les noms de fichier. Les noms de membre de fichier PDS peuvent comporter jusqu'à huit caractères et ne peuvent pas contenir de point (.). Lorsque vous effectuez un transfert vers un fichier, vous devez spécifier explicitement le nom, ce qui signifie que ces restrictions de dénomination ne posent pas de problème. Toutefois, lorsque vous effectuez un transfert depuis des fichiers vers des membres PDS, le chemin d'accès au fichier peut ne pas être mappé vers un nom de membre PDS. Lorsque vous effectuez un transfert vers un fichier PDS, chaque fichier source devient un membre PDS et chaque nom de membre est généré à partir du nom de la source.

Les noms de membre PDS sont des noms z/OS non qualifiés et sont définis par l'expression régulière suivante:

```
[a-zA-Z$#@] [a-zA-Z0-9$#@]{0-7}
```

Le schéma suivant permet de convertir un nom de fichier source ou de fichier source en nom de membre PDS valide. Les considérations sont appliquées dans l'ordre indiqué:

1. Seuls les caractères du nom après la dernière barre oblique (/), la dernière barre oblique inversée (\) ou le dernier signe deux-points (:) sont utilisés. C'est-à-dire que seule la partie nom d'un chemin de fichier est utilisée.
2. Pour les fichiers source (qui ne sont pas des ensembles de données ou des membres PDS), les caractères après et incluant le dernier point (.) sont ignorés.
3. Pour tout nom de plus de huit caractères, seuls les huit derniers caractères sont utilisés.
4. Les caractères point sont remplacés par des caractères arobase (@).
5. Les caractères non valides sont remplacés par des caractères arobase (@).
6. Si la conversion ne génère aucun caractère, le nom du membre PDS est @.

Messages d'état de l'agent MFT

Les agents de haute disponibilité publient les informations de statut au format XML.

Exemple de code XML présentant des informations sur trois instances de secours

```
<?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>
```

Publication du statut de l'agent avec le statut de secours XML imbriqué.

Le code XML de statut de secours est affiché en gras.

```
<?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" /
```

```
<entry key="agentVersion">9.1.4.0</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"?&lt;
  agentTraceStatus version="6.00" xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:noNamespaceSchemaLocation="AgentTraceStatus.xsd"&lt;
  level="all"&lt;
  &lt;/agentTraceStatus&lt;
</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>
```

Référence associée

[fteCreateAgent](#)

[agent GET](#)

IBM MQ Internet Pass-Thru Référence de configuration

IBM MQ Internet Pass-Thru (MQIPT) utilise un fichier de configuration appelé `mqipt.conf` pour définir les routes et contrôler les actions du serveur MQIPT. Depuis la IBM MQ 9.2, les propriétés de configuration de la commande `mqiptAdmin` peuvent également être spécifiées dans un fichier de propriétés.

Le fichier de configuration MQIPT

Le fichier de configuration MQIPT comprend un certain nombre de sections. Il existe une section `[global]` et une section `[route]` supplémentaire pour chaque route via MQIPT qui a été définie.

Chaque section contient des paires de propriétés nom/valeur. Certaines propriétés n'apparaissent que dans la section `[global]`, d'autres uniquement dans la section `[route]`, et certaines dans les sections `[route]` et `[global]`. Si une propriété apparaît à la fois dans les sections `[route]` et `[global]`, la valeur de la propriété dans la section `[route]` remplace la valeur globale, mais uniquement pour la route en question. De cette manière, la section `[global]` peut être utilisée pour établir les valeurs par défaut à utiliser pour les propriétés qui ne sont pas définies dans les sections `[route]` individuelles.

La section `[global]` commence par une ligne contenant les caractères `[global]` et se termine à l'endroit où commence la première section `[route]`. La section `[global]` doit précéder toutes les sections `[route]` dans le fichier.

Chaque section `[route]` commence par une ligne contenant les caractères `[route]` et se termine à l'endroit où commence la première section `[route]`, ou lorsque la fin du fichier de configuration est atteinte.

Tout nom de propriété non reconnu est ignoré. Si une propriété d'une section `[route]` a un nom reconnu mais une valeur incorrecte (par exemple `MinConnectionThreads=x` ou `HTTP=unsure`), cette route est désactivée (c'est-à-dire qu'elle n'écoute pas les connexions entrantes).



Avertissement : La limite maximale du nombre de routes pouvant être ajoutées dans le fichier `mqipt.conf` est de 100.

Des valeurs non valides pour les propriétés de la section [global] peuvent empêcher le démarrage de MQIPT ou du serveur de commandes. Si le serveur de commandes ne démarre pas, MQIPT n'écoute pas les commandes d'administration envoyées par la commande **mqiptAdmin** au port de commande affecté. Si des propriétés avec des valeurs non valides dans la section [global] sont présentes lorsque MQIPT est actualisé, un message d'avertissement est émis et la valeur effective de la propriété reste inchangée. Cela empêche les valeurs de propriété non valides de provoquer l'arrêt d'une instance active de MQIPT lorsqu'elle est actualisée.

Lorsqu'une propriété est répertoriée comme prenant les valeurs true ou false, tout mélange de majuscules et de minuscules peut être utilisé dans la valeur de la propriété.

Vous pouvez modifier la valeur d'une propriété en éditant le fichier `mqipt.conf`. Pour appliquer les modifications, actualisez MQIPT à l'aide de la commande **mqiptAdmin** avec le mot clé **-refresh**.

Pour inclure des commentaires dans le fichier de configuration, commencez par une ligne avec le caractère "#".

Les modifications de certaines propriétés ne déclenchent le redémarrage d'une route que si d'autres propriétés sont déjà activées. Par exemple, toutes les modifications apportées aux propriétés HTTP ne prendront effet que si la propriété **HTTP** est aussi activée.

Lorsqu'une route est redémarrée, les connexions existantes sont arrêtées. Pour annuler ce comportement, définissez la propriété **RouteRestart** sur false. Cela empêche la route de redémarrer, ce qui permet aux connexions existantes de rester actives jusqu'à ce que la propriété **RouteRestart** soit réactivée.

Pour plus d'informations sur la configuration de certaines configurations simples, voir [Initiation à MQIPT](#). Pour obtenir un exemple de configuration, voir le fichier `mqiptSample.conf` dans le répertoire d'installation de MQIPT.

Le fichier de propriétés mqiptAdmin

Les propriétés de configuration de la commande **mqiptAdmin** peuvent être spécifiées dans un fichier de propriétés distinct. Ces propriétés de configuration sont nécessaires lorsque **mqiptAdmin** se connecte au port de commande TLS MQIPT.

Pour obtenir la liste des propriétés pouvant être spécifiées dans le fichier de propriétés **mqiptAdmin**, voir «[mqiptAdminpropriétés](#)», à la page 275. Les noms de propriété sont sensibles à la casse. Toutes les propriétés non reconnues sont ignorées.

Les commentaires peuvent être inclus dans le fichier de propriétés en démarrant une ligne par un caractère "#".

Récapitulatif des propriétés MQIPT

Le tableau ci-dessous affiche un récapitulatif des propriétés de configuration de MQIPT et inclut les informations suivantes :

- Liste alphabétique de propriétés de MQIPT avec des liens vers d'autres informations dans la section [route], ou la section [global] si la section [route] ne s'applique pas.
- Indique si la propriété doit avoir la valeur true pour entrer en vigueur.
- Indique si la propriété s'applique à la section [global], [route] ou aux deux.
- Valeurs par défaut utilisées si une propriété est manquante dans la section [route] et la section [global]. Lorsque vous spécifiez les valeurs true et false, toute combinaison de caractères en majuscules et en minuscules peut être utilisée.



Nom de la propriété	Propriété définie sur true	Global	Route	Par défaut
AccessPW		oui	non	null

Nom de la propriété	Propriété définie sur true	Global	Route	Par défaut
Actif		oui	oui	true
V 9.4.0 V 9.4.0 «[MQ 9.4.0 Juin 2024][MQ 9.4.0 Juin 2024]AllowedProtocols», à la page 254		oui	oui	mq
ClientAccess		oui	oui	faux
CommandPort		oui	non	null
CommandPortListenerAddress		oui	non	null
ConnectionLog		oui	non	true
Destination		non	oui	null
DestinationPort		non	oui	1414
«Fonctions EnableAdvanced», à la page 252		oui	non	faux
HTTP		oui	oui	faux
V 9.4.0 V 9.4.0 «[MQ 9.4.0 Juin 2024][MQ 9.4.0 Juin 2024]HTTPConnectionTimeout», à la page 255		oui	oui	5000
HTTPProxy	HTTP	oui	oui	null
HTTPProxyPort	HTTP	oui	oui	8080
HTTPS	HTTP	oui	oui	faux
HTTPServer	HTTP	oui	oui	null
HTTPServerPort	HTTP	oui	oui	null
IdleTimeout		oui	oui	0
IgnoreExpiredCRLs		oui	oui	faux
Protocole LDAP		oui	oui	faux
LDAPIgnoreErrors	LDAP	oui	oui	faux
LDAPCacheTimeout	LDAP	oui	oui	24
LDAPServer1	LDAP	oui	oui	null
LDAPServer1Port	LDAP	oui	oui	389
LDAPServer1Userid	LDAP	oui	oui	null
LDAPServer1Password	LDAP	oui	oui	null
LDAPServer1Timeout	LDAP	oui	oui	0
LDAPServer2	LDAP	oui	oui	null
LDAPServer2Port	LDAP	oui	oui	389
LDAPServer2Userid	LDAP	oui	oui	null
LDAPServer2Password	LDAP	oui	oui	null


Nom de la propriété	Propriété définie sur true	Global	Route	Par défaut
LDAPServer2Timeout	LDAP	oui	oui	0
ListenerAddress		oui	oui	null
ListenerPort		non	oui	null
LocalAddress		oui	oui	null
LocalAdmin		oui	non	true
MaxConnectionThreads		oui	oui	100
MaxLogFileSize		oui	non	50
MinConnectionThreads		oui	oui	5
Nom		non	oui	null
OutgoingPort		non	oui	0
V 9.4.0 PasswordProtection		oui	oui	required
QMgrAccess		oui	oui	true
AuthenticationRemoteCommand		oui	non	Aucun
RemoteShutdown		oui	non	faux
RouteRestart		oui	oui	true
SecurityExit		oui	oui	faux
SecurityExitName	SecurityExit	oui	oui	null
SecurityExitPath	SecurityExit	oui	oui	<i>mqi</i> pt_home \\exits
SecurityExitTimeout	SecurityExit	oui	oui	30
SecurityManager (Remarque 3)		oui	non	faux
StratégieSecurityManager (Remarque 3)		oui	non	null
SocksClient		oui	oui	faux
SocksProxyHost	SocksClient	oui	oui	null
SocksProxyPort	SocksClient	oui	oui	1080
SocksServer		oui	oui	faux
SSLClient		oui	oui	faux
SSLClientCAKeyRing	SSLClient	oui	oui	null
SSLClientCAKeyRingPW	SSLClient	oui	oui	null
« SSLClientCAKeyRingUseCryptoHardware », à la page 261	SSLClient	oui	oui	faux
SSLClientCipherSuites	SSLClient	oui	oui	null
SSLClientConnectTimeout	SSLClient	oui	oui	30
SSLClientCustomOutboundSNI	SSLClient	oui	oui	null

Nom de la propriété	Propriété définie sur true	Global	Route	Par défaut
SSLClientDN_C	SSLClient	oui	oui	* (Note 1)
SSLClientDN_CN	SSLClient	oui	oui	* (Note 1)
SSLClientDN_DC	SSLClient	oui	oui	* (Note 1)
SSLClientDN_DNQ	SSLClient	oui	oui	* (Note 1)
SSLClientDN_L	SSLClient	oui	oui	* (Note 1)
SSLClientDN_O	SSLClient	oui	oui	* (Note 1)
SSLClientDN_OU	SSLClient	oui	oui	* (Note 1)
SSLClientDN_PC	SSLClient	oui	oui	* (Note 1)
SSLClientDN_ST	SSLClient	oui	oui	* (Note 1)
SSLClientDN_Street	SSLClient	oui	oui	* (Note 1)
SSLClientDN_T	SSLClient	oui	oui	* (Note 1)
SSLClientDN_UID	SSLClient	oui	oui	* (Note 1)
SSLClientExit		oui	oui	faux
SSLClientKeyRing	SSLClient	oui	oui	null
SSLClientKeyRingPW	SSLClient	oui	oui	null
« SSLClientKeyRingUseCryptoHardware », à la page 264	SSLClient	oui	oui	faux
« SNI SSLClientOutbound », à la page 264	SSLClient	oui	oui	nom_hôte
SSLClientProtocols	SSLClient	oui	oui	TLSv1.2 TLSv1.3
SSLClientSiteDN_C	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_CN	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_DC	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_DNQ	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_L	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_O	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_OU	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_PC	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_ST	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_Street	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_T	SSLClient	oui	oui	* (Note 1)
SSLClientSiteDN_UID	SSLClient	oui	oui	* (Note 1)
SSLClientSiteLabel	SSLClient	oui	oui	null
SSLCommandPort		oui	non	null
SSLCommandPortCipherSuites		oui	non	null

Nom de la propriété	Propriété définie sur <code>true</code>	Global	Route	Par défaut
SSLCommandPortListenerAddress		oui	non	null
SSLCommandPortKeyRing		oui	non	null
SSLCommandPortKeyRingPW		oui	non	null
SSLCommandPortKeyRingUseCryptoMaterial		oui	non	faux
ProtocolesSSLCommandPort		oui	non	TLSv1.2 TLSv1.3
SSLCommandPortSiteLabel		oui	non	null
SSLExitData	SSLServerExit	oui	oui	null
SSLExitName	SSLServerExit	oui	oui	null
SSLExitPath	SSLServerExit	oui	oui	<i>mqipt_home</i> \ exits
SSLExitTimeout	SSLServerExit	oui	oui	30
SSLProxyMode		oui	oui	faux
SSLPlainConnections	either SSLServer or SSLProxyMode	oui	oui	faux
SSLServer		oui	oui	faux
SSLServerAskClientAuth	SSLServer	oui	oui	faux
SSLServerCAKeyRing	SSLServer	oui	oui	null
SSLServerCAKeyRingPW	SSLServer	oui	oui	null
« SSLServerCAKeyRingUseCryptoHardware », à la page 269	SSLServer	oui	oui	faux
SSLServerCipherSuites	SSLServer	oui	oui	null
SSLServerDN_C	SSLServer	oui	oui	* (Note 1)
SSLServerDN_CN	SSLServer	oui	oui	* (Note 1)
SSLServerDN_DC	SSLServer	oui	oui	* (Note 1)
SSLServerDN_DNQ	SSLServer	oui	oui	* (Note 1)
SSLServerDN_L	SSLServer	oui	oui	* (Note 1)
SSLServerDN_O	SSLServer	oui	oui	* (Note 1)
SSLServerDN_OU	SSLServer	oui	oui	* (Note 1)
SSLServerDN_PC	SSLServer	oui	oui	* (Note 1)
SSLServerDN_ST	SSLServer	oui	oui	* (Note 1)
SSLServerDN_Street	SSLServer	oui	oui	* (Note 1)
SSLServerDN_T	SSLServer	oui	oui	* (Note 1)
SSLServerDN_UID	SSLServer	oui	oui	* (Note 1)
SSLServerExit		oui	oui	faux

Nom de la propriété	Propriété définie sur true	Global	Route	Par défaut
SSLServerKeyRing	SSLServer	oui	oui	null
SSLServerKeyRingPW	SSLServer	oui	oui	null
« SSLServerKeyRingUseCryptoHardware », à la page 271	SSLServer	oui	oui	faux
SSLServerProtocols	SSLServer	oui	oui	TLSv1.2 TLSv1.3
SSLServerSiteDN_C	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_CN	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_DC	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_DNQ	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_L	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_O	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_OU	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_PC	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_ST	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_Street	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_T	SSLServer	oui	oui	* (Note 1)
SSLServerSiteDN_UID	SSLServer	oui	oui	* (Note 1)
SSLServerSiteLabel	SSLServer	oui	oui	null
FormatStoredCredentials		oui	oui	null
TCPKeepAlive		oui	oui	false
Trace		oui	oui	0
 TraceFileNombre		oui	non	25
 TraceFileTaille		oui	non	200
«Données TraceUser », à la page 274		oui	oui	64
UriName	HTTP	oui	oui	(Note 2)

Remarques :

1. L'astérisque (*) représente un caractère générique.
2. Voir [UriName](#) dans «MQIPT Propriétés de la route», à la page 254 pour plus d'informations sur les paramètres par défaut.
3.  Cette propriété est obsolète pour être supprimée dans une édition ultérieure.

Référence associée

«IBM MQ Internet Pass-Thru Référence de configuration», à la page 244

IBM MQ Internet Pass-Thru (MQIPT) utilise un fichier de configuration appelé `mqipt.conf` pour définir les routes et contrôler les actions du serveur MQIPT. Depuis la IBM MQ 9.2, les propriétés de configuration de la commande **mqiptAdmin** peuvent également être spécifiées dans un fichier de propriétés.

«MQIPT propriétés globales», à la page 251

Le fichier de configuration `mqipt.conf` peut contenir un certain nombre de propriétés globales.

«MQIPT Propriétés de la route», à la page 254

Le fichier de configuration `mqipt.conf` peut contenir les propriétés de chacune des routes.

MQIPT propriétés globales

Le fichier de configuration `mqipt.conf` peut contenir un certain nombre de propriétés globales.

Les propriétés suivantes ne peuvent apparaître que dans la section `[global]` de `mqipt.conf`.

Toutes les propriétés de route, sauf **ListenerPort**, **Destination**, **DestinationPort**, **Name** et **OutgoingPort** peuvent aussi figurer dans la section `[global]`. Si une propriété apparaît à la fois dans les sections `route` et `[global]`, la valeur de la propriété dans la section `[route]` remplace la valeur globale, mais uniquement pour la route en question. De cette manière, la section `[global]` peut être utilisée pour établir les valeurs par défaut à utiliser pour les propriétés qui ne sont pas définies dans les sections `[route]` individuelles.

AccessPW

Mot de passe utilisé pour authentifier les commandes envoyées au port de commande MQIPT à l'aide de la commande `mqiptAdmin`.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande `mqiptPW`, soit un mot de passe en texte en clair. Les mots de passe en texte en clair ne peuvent contenir que des caractères alphanumériques. Il est fortement recommandé de chiffrer les mots de passe stockés dans la configuration MQIPT. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir Chiffrement des mots de passe stockés.

L'authentification est effectuée pour les commandes d'administration reçues par le port de commande si les deux conditions suivantes sont remplies:

- La propriété **AccessPW** est spécifiée et définie sur une valeur qui n'est pas vide.
- La propriété **RemoteCommandAuthentication** est spécifiée et définie sur une valeur autre que `none`.

CommandPort

Numéro de port TCP/IP du port de commande non sécurisé. MQIPT accepte les commandes d'administration envoyées par la commande `mqiptAdmin` à ce port de commande.

Les connexions au port de commande non sécurisé ne sont pas sécurisées avec TLS. Les données envoyées au port de commande, y compris le mot de passe d'accès, peuvent être accessibles par d'autres utilisateurs du réseau. Pour configurer un port de commande sécurisé avec TLS, définissez la propriété **SSLCommandPort** à la place.

Si la propriété **CommandPort** n'est pas spécifiée, MQIPT n'écoute pas les commandes d'administration sur le port de commande non sécurisé. Pour utiliser le numéro de port par défaut, 1881, utilisé par défaut par la commande `mqiptAdmin`, définissez **CommandPort** sur 1881.

CommandPortListenerAddress

Adresse du programme d'écoute local à utiliser par le port de commande non sécurisé. En définissant l'adresse du programme d'écoute local, vous pouvez limiter les connexions entrantes vers le port de commande non sécurisé à celles provenant d'une interface réseau particulière. La valeur par défaut est d'écouter sur toutes les interfaces réseau.

ConnectionLog

Les valeurs admises sont `true` ou `false`. Lorsque la valeur est `true`, MQIPT consigne toutes les tentatives de connexion (réussies ou non) dans le sous-répertoire `logs` et les événements de déconnexion dans le fichier `mqiptYYYYMMDDHHmmSS.log` (où `YYYYMMDDHHmmSS` sont des caractères représentant la date et l'heure en cours). La valeur par défaut de **ConnectionLog** est `true`. Lorsque la valeur `true` de cette propriété est remplacée par `false`, MQIPT ferme le journal de connexion existant et en crée un nouveau. Le nouveau journal est utilisé lorsque la propriété est redéfinie comme `true`.

Fonctions EnableAdvanced

Définissez cette propriété sur `true` pour confirmer que les fonctions avancées qui requièrent une autorisation d'utilisation IBM MQ Advanced, IBM MQ Advanced for z/OS, IBM MQ Advanced for z/OS VUE, ou IBM MQ Appliance peuvent être utilisées par MQIPT. Si vous disposez des droits appropriés, vous pouvez utiliser les fonctions avancées de MQIPT. Si des fonctions avancées sont activées sur une route, le gestionnaire de files d'attente local qui est connecté à l'aide de la route MQIPT doit également disposer des droits IBM MQ Advanced, IBM MQ Advanced for z/OS, IBM MQ Advanced for z/OS VUE, ou IBM MQ Appliance. Les routes qui utilisent des fonctions avancées ne peuvent pas démarrer sauf si cette propriété est définie sur `true`. Lorsque cette propriété passe de `true` à `false`, les routes qui utilisent les fonctions avancées sont arrêtées.

LocalAdmin

Indique si l'administration locale sans port de commande est autorisée. Les commandes d'administration envoyées par la commande `mqiptAdmin` à l'aide de l'administration locale à la place du port de commande ne sont pas acceptées si cette propriété est définie sur `false`.

Les valeurs valides pour cette propriété sont `true` et `false`. La valeur par défaut est `true`.

MaxLogFileSize

La taille maximale (exprimé en Ko) du fichier journal de la connexion. Lorsque ce fichier dépasse cette taille maximale, une copie de sauvegarde (`mqipt001.log`) est effectuée, et un nouveau fichier est démarré. Seuls deux fichiers de sauvegarde sont conservés (`mqipt001.log` et `mqipt002.log`). Chaque fois que le fichier journal principal est saturé, les sauvegardes antérieures sont effacées. La valeur par défaut de **MaxLogFileSize** est 50 ; la valeur minimale autorisée est 5.

Authentification RemoteCommand

Indique si les commandes d'administration reçues par le port de commande non sécurisé ou le port de commande TLS doivent être authentifiées. Les commandes sont authentifiées en vérifiant que le mot de passe fourni correspond au mot de passe spécifié dans la propriété `AccessPW`. Les valeurs admises sont les suivantes :

aucun

Aucune authentification n'est effectuée sur les commandes émises vers l'un ou l'autre des ports de commande. Les utilisateurs de la commande `mqiptAdmin` n'ont pas besoin d'entrer de mot de passe. Il s'agit de la valeur par défaut.

facultatif

Les utilisateurs de la commande `mqiptAdmin` ne sont pas tenus de fournir un mot de passe. Toutefois, si un mot de passe est fourni, il doit être valide.

obligatoire

Les utilisateurs de la commande `mqiptAdmin` doivent fournir un mot de passe valide pour chaque commande émise sur les ports de commande.

La propriété **AccessPW** doit également être spécifiée pour activer l'authentification pour les ports de commande.

RemoteShutDown

Indique si MQIPT peut être arrêté par une commande d'arrêt envoyée au port de commande non sécurisé ou au port de commande TLS par la commande `mqiptAdmin`. Cette propriété doit être définie sur `true` pour les commandes d'arrêt reçues par l'un des ports de commande à traiter.

Les valeurs valides pour cette propriété sont `true` et `false`. La valeur par défaut est `false`.

SecurityManager

Associez cette propriété à la valeur `true` pour activer Java security manager pour cette instance de MQIPT. Vous devez vérifier que les droits adéquats sont accordés. Pour plus d'informations, voir [Java security manager](#). La valeur par défaut de cette propriété est `false`.

Cette propriété est obsolète pour être supprimée dans une édition ultérieure.

SecurityManagerPolicy

Nom de fichier qualifié complet d'un fichier de règles Java security manager. Si cette propriété n'est pas définie, seuls les fichiers de règles système et utilisateur par défaut sont utilisés. Si le Java security manager est déjà activé, les modifications apportées à cette propriété n'ont aucun effet tant que le Java security manager n'a pas été désactivé et réactivé.

Deprecated Cette propriété est obsolète pour être supprimée dans une édition ultérieure.

SSLCommandPort

Numéro de port TCP/IP du port de commande TLS. MQIPT accepte les commandes d'administration envoyées par la commande **mqiptAdmin** à ce port de commande. Ce port accepte uniquement les connexions TLS. Cette propriété doit être spécifiée pour activer le port de commande TLS.

SSLCommandPortCipherSuites

Nom des suites de chiffrement à activer sur le port de commande TLS. Plusieurs suites de chiffrement peuvent être spécifiées en séparant les valeurs par des virgules. Seules les suites de chiffrement TLS 1.2 et TLS 1.3 qui sont activées par défaut dans le Java runtime environment (JRE) fourni avec MQIPT peuvent être spécifiées. Si cette propriété n'est pas spécifiée, toutes les suites de chiffrement qui sont activées dans l'environnement d'exécution Java sont activées sur le port de commande TLS.

SSLCommandPortListenerAddress

Adresse du programme d'écoute local à utiliser par le port de commande TLS. En définissant l'adresse du programme d'écoute local, vous pouvez restreindre les connexions entrantes au port de commande TLS à celles provenant d'une interface réseau particulière. La valeur par défaut est d'écouter sur toutes les interfaces réseau.

SSLCommandPortKeyRing

Nom du fichier de clés PKCS#12 qui contient le certificat du serveur de port de commande TLS.

Sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\\) comme séparateur de fichier.

SSLCommandPortKeyRingPW

Mot de passe chiffré permettant d'accéder au fichier de clés du port de commande TLS ou au magasin de clés PKCS #11 . Le mot de passe doit être chiffré à l'aide de la commande **mqiptPW** et la valeur de cette propriété doit être définie sur la chaîne générée par **mqiptPW**.

SSLCommandPortKeyRingUseCryptoMatériel

Indique si le matériel de cryptographie qui prend en charge l'interface PKCS #11 est utilisé comme magasin de clés pour le certificat du serveur de port de commande TLS. Les valeurs valides pour cette propriété sont **true** et **false**. Si cette propriété est définie sur **true**, **SSLCommandPortKeyRing** ne peut pas être également spécifié.

L'utilisation du matériel de cryptographie dans MQIPT est une fonction IBM MQ Advanced . La propriété **EnableAdvancedCapabilities** doit être définie sur **true** pour confirmer que vous disposez des droits IBM MQ Advanced .

Protocoles SSLCommandPort

Liste de protocoles séparés par des virgules à activer sur le port de commande TLS. Une ou plusieurs des valeurs suivantes peuvent être spécifiées.

<i>Tableau 117. Valeurs autorisées pour les protocoles TLS de port de commande</i>	
Valeur	Protocole
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

Si vous ne spécifiez pas cette propriété, TLS 1.2 et TLS 1.3 sont activés par défaut.

SSLCommandPortSiteLabel

Nom de libellé du certificat serveur utilisé par le port de commande TLS. Si cette propriété n'est pas spécifiée, tout certificat du magasin de clés du port de commande TLS compatible avec la suite de chiffrement est sélectionné.

Fonction de trace

Le niveau de trace des unités d'exécution globale MQIPT qui ne sont pas associées à une route, et des routes qui n'ont pas de propriété **Trace** définie. Par exemple, l'unité d'exécution de contrôle MQIPT principale et les unités d'exécution du serveur de commandes ne sont pas associées à une route et

ne sont tracées que si la trace est activée dans la section [global] . La valeur de la propriété **Trace** dans une section [route] remplace la propriété **Trace** de cette route. Pour plus d'informations sur le traçage des unités d'exécution associées à une route, voir **Trace** dans la section [route].

La valeur de cette propriété peut être l'une des suivantes:

0

La trace n'est pas activée

Tout entier positif

La trace est activée

La valeur par défaut est 0.

V 9.4.0 TraceFileNombre

Nombre de fichiers de trace dans l'ensemble tournant de fichiers utilisés par MQIPT pour écrire des données de trace.

La valeur minimale autorisée est 3. La valeur par défaut est 25.

Si vous modifiez la valeur de cette propriété, le fichier de trace en cours est fermé et le fichier suivant de l'ensemble tournant de fichiers de trace est ouvert.

V 9.4.0 Taille de TraceFile

Taille maximale des fichiers de trace produits par MQIPT, spécifiée en Mo.

La valeur minimale autorisée est 1. La valeur par défaut est 200.

Si vous modifiez la valeur de cette propriété, le fichier de trace en cours est fermé et le fichier suivant de l'ensemble tournant de fichiers de trace est ouvert.

MQIPT Propriétés de la route

Le fichier de configuration mqipt.conf peut contenir les propriétés de chacune des routes.

La section [route] du fichier de configuration mqipt.conf peut contenir les propriétés suivantes :

Actif

La route autorise les connexions entrantes uniquement si la propriété **Active** a pour valeur `true`. Cela signifie que vous pouvez interrompre temporairement l'accès à la destination, en définissant cette valeur sur `false`, sans devoir supprimer la section [route] du fichier de configuration. Si vous définissez cette propriété sur `false`, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

V 9.4.0 **V 9.4.0** AllowedProtocols

Indique les protocoles qui sont acceptés par cette route. Les connexions qui utilisent un protocole qui n'est pas spécifié dans cette liste sont rejetées. Plusieurs protocoles peuvent être spécifiés sous la forme d'une liste séparée par des virgules. Les protocoles suivants peuvent être spécifiés:

mq

La route accepte les connexions qui utilisent le protocole IBM MQ .

http

La route accepte les connexions HTTP provenant d'une autre instance de MQIPT.

La valeur par défaut de cette propriété est `mq`.

Si la valeur de cette propriété est modifiée, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

ClientAccess

La route autorise les connexions de canal client entrantes uniquement si la valeur de **ClientAccess** est `true`. Vous pouvez configurer MQIPT afin qu'il accepte les demandes client uniquement, les

demandes de gestionnaires de files d'attente uniquement, ou les deux. Utilisez cette propriété en conjonction avec la propriété **QMgrAccess**. Si vous définissez cette propriété sur `false`, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

Destination

Le nom d'hôte (ou l'adresse IP à notation décimale) du gestionnaire de files d'attente ou l'instance de MQIPT suivante, auquel/à laquelle cette route doit se connecter. Chaque section [route] doit contenir une valeur explicite **Destination**, mais plusieurs sections [route] peuvent désigner la même destination. Si une modification de cette propriété affecte cette route, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées. Lors de l'utilisation de la propriété **SocksProxyHost**, la propriété **Destination** doit utiliser le format d'adresse IPv4 en notation décimale à point.

DestinationPort

Port de l'hôte de destination auquel cette route doit se connecter. Chaque section [route] doit contenir une valeur explicite **DestinationPort**, mais plusieurs routes peuvent faire référence à la même combinaison de valeurs **Destination** et **DestinationPort**. Si une modification de cette propriété affecte cette route, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

HTTP

Définissez **HTTP** sur `true` pour les routes chargées d'envoyer des demandes de tunnellation HTTP sortantes. La propriété **Destination** de la route doit être le nom d'hôte d'une autre instance de MQIPT lorsque **HTTP** est défini sur `true`. Définissez **HTTP** sur `false` pour les routes connectées à des gestionnaires de files d'attente IBM MQ. Si vous modifiez cette propriété, la route est arrêtée. Au moins une des propriétés **HTTPProxy** ou **HTTPServer** doit également être spécifiée lorsque **HTTP** est défini sur `true`. Cette propriété ne peut pas être utilisée conjointement avec la propriété **SocksClient**.

V 9.4.0

V 9.4.0

HTTPConnectionTimeout

Cette propriété indique le temps, exprimé en millisecondes, pendant lequel MQIPT attend qu'une connexion HTTP reçue soit correctement établie, avant que la connexion ne soit rejetée.

La valeur par défaut est 5000.

HTTPProxy

Le nom d'hôte (ou adresse IP à notation décimale) du proxy HTTP utilisé par toutes les connexions de cette route. Une demande **CONNECT** est envoyée au proxy HTTP, au lieu de la demande **POST** normalement utilisée lorsqu'aucun proxy HTTP n'est configuré. Si vous modifiez cette propriété (et si **HTTP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

HTTPProxyPort

L'adresse du port à utiliser sur le proxy HTTP. La valeur par défaut est 8080. Si vous modifiez cette propriété (et si **HTTP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

HTTPServer

Le nom d'hôte (ou adresse IP à notation décimale) du serveur HTTP utilisé par toutes les connexions de cette route. Il s'agit généralement du nom d'hôte d'un autre MQIPT.

Si **HTTPProxy** n'est pas spécifié, MQIPT se connecte à l'hôte spécifié dans **HTTPServer** et envoie des demandes HTTP **POST** à l'hôte spécifié dans la propriété **Destination** de la route. Si **HTTPProxy** est spécifié, MQIPT se connecte à l'hôte spécifié dans **HTTPProxy** à la place et demande que le proxy établisse un tunnel vers l'hôte spécifié dans **HTTPServer**.

Si **HTTPProxy** est spécifié la valeur par défaut est la route **Destination**.

Si vous modifiez cette propriété (et si **HTTP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

HTTPS

Définissez **HTTPS** sur `true` pour émettre des demandes HTTPS. Les propriétés **HTTP** et **SSLClient** doivent également être activées, et le fichier de clés du client doit être configuré à l'aide de

la propriété **SSLClientKeyRing** ou **SSLClientKeyRingUseCryptoHardware**, comme pour l'opération SSL/TLS. Si vous modifiez la propriété **HTTPS** (et si **HTTP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

HTTPServerPort

L'adresse du port à utiliser sur le serveur HTTP. La valeur par défaut est 8080, sauf si la propriété **HTTPProxy** est spécifiée, auquel cas la valeur par défaut est la route **DestinationPort**.

Si vous modifiez cette propriété (et si **HTTP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

IdleTimeout

Le délai, en minutes, au terme duquel une connexion inactive est fermée. Les canaux du gestionnaire de files d'attente ont aussi la propriété **DISCINT**. Si vous définissez le paramètre **IdleTimeout**, n'oubliez pas que **DISCINT** est utilisé. Si **IdleTimeout** est défini sur 0, il n'y a aucun délai d'inactivité. Les modifications apportées à cette propriété prennent effet uniquement lorsque la route est redémarrée.

IgnoreExpiredCRLs

Associez la propriété **IgnoreExpiredCRLs** à la valeur `true` pour ignorer une liste de révocation de certificat (CRL) arrivée à expiration. La valeur par défaut est `false`. Notez que si vous associez la propriété **IgnoreExpiredCRLs** à la valeur `true`, il se peut qu'un certificat révoqué soit utilisé pour établir une connexion SSL/TLS.

LDAP

Associez la propriété **LDAP** à la valeur `true` pour activer l'utilisation d'un serveur LDAP lors de l'utilisation de connexions SSL/TLS. MQIPT utilise le serveur LDAP pour extraire des listes CRL et ARL. La propriété **SSLClient** ou **SSLServer** doit également avoir pour valeur `true` pour que cette propriété entre en vigueur.

LDAPCacheTimeout

Le délai d'expiration, exprimé en heures, du cache temporaire dans lequel une liste CRL extraite à partir d'un serveur LDAP est stockée. Une fois ce délai écoulé, la totalité du cache CRL est vidée. Par exemple, si vous indiquez une valeur de 1 heure, le cache est vidé une fois par heure. La valeur par défaut est 24. Si vous spécifiez une valeur de délai de 0, les entrées contenues dans le cache n'arrivent pas à expiration tant que la route n'a pas redémarré. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPIgnoreErrors

Définissez **LDAPIgnoreErrors** sur `true` pour ignorer les erreurs de connexion ou de délai d'attente lors de l'exécution d'une recherche LDAP. Si MQIPT ne peut pas effectuer de recherche, il n'autorise pas la connexion client à se terminer, sauf si cette propriété a été activée. Une recherche a réussi lorsqu'une liste CRL a été extraite ou il n'existe pas de certificat CRL disponible pour l'autorité de certification indiquée. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

Remarque : Si vous activez cette propriété, il se peut qu'un certificat révoqué soit utilisé pour établir une connexion SSL/TLS.

LDAPServer1

Le nom d'hôte ou l'adresse IP du serveur LDAP principal. Cette propriété doit être définie si **LDAP** a été défini sur `true`. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer1Port

Le numéro de port d'écoute du serveur LDAP principal. La valeur par défaut est 389. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer1Userid

L'ID utilisateur nécessaire pour accéder au serveur LDAP principal. Cette propriété doit être définie si l'autorisation d'accès au serveur LDAP principal est requise. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer1Password

Le mot de passe nécessaire pour accéder au serveur LDAP principal. Cette propriété doit être définie si **LDAPServer1Userid** a été défini sur `true`. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande **mqiptPW**, soit un mot de passe en texte en clair. Les mots de passe en texte en clair ne peuvent contenir que des caractères alphanumériques. Il est fortement recommandé de chiffrer les mots de passe stockés dans la configuration MQIPT. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir [Chiffrement des mots de passe stockés](#).

LDAPServer1Timeout

Le délai, en secondes, pendant lequel MQIPT attend une réponse du serveur LDAP principal. La valeur par défaut est 0, ce qui signifie que la connexion n'expire pas. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer2

Le nom d'hôte ou l'adresse IP du serveur LDAP de secours. Cette propriété est facultative. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer2Port

Le numéro de port d'écoute du serveur LDAP de secours. La valeur par défaut est 389. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer2Userid

L'ID utilisateur nécessaire pour accéder au serveur LDAP de secours. Cette propriété doit être définie si l'autorisation d'accès au serveur LDAP de secours est requise. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

LDAPServer2Password

Le mot de passe nécessaire pour accéder au serveur LDAP de secours. Cette propriété doit être définie si **LDAPServer2** a été défini sur `true`. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande **mqiptPW**, soit un mot de passe en texte en clair. Les mots de passe en texte en clair ne peuvent contenir que des caractères alphanumériques. Il est fortement recommandé de chiffrer les mots de passe stockés dans la configuration MQIPT. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir [Chiffrement des mots de passe stockés](#).

LDAPServer2Timeout

Le délai, en secondes, pendant lequel MQIPT attend une réponse du serveur LDAP de secours. La valeur par défaut est 0, ce qui signifie que la connexion n'expire pas. Si vous modifiez cette propriété (et que **LDAP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

ListenerAddress

Utilisez cette propriété si le système MQIPT possède plusieurs adresses IP et si vous devez lier le port d'écoute de la route à une adresse spécifique. Cette option est utile pour limiter les connexions entrantes à celles d'une interface réseau donnée. La valeur de cette propriété doit être une adresse IP appartenant à l'une des interfaces réseau sur le système où s'exécute MQIPT. La valeur par défaut consiste à accepter les connexions provenant de toutes les interfaces réseau.

ListenerPort

Le numéro de port sur lequel la route doit écouter les demandes entrantes. Chaque section [route] doit contenir une valeur explicite **ListenerPort**. Les valeurs de **ListenerPort** définies dans chaque section doivent être différentes. N'importe quel numéro de port valide peut être utilisé, y compris les ports 80 et 443, à condition que les ports choisis ne soient pas déjà utilisés par un autre programme d'écoute TCP/IP s'exécutant sur le même hôte.

LocalAddress

L'adresse IP permettant de lier toutes les connexions à cette route sur cet ordinateur. L'adresse choisie doit être une adresse IP associée à l'une des interfaces réseau sur l'ordinateur sur lequel s'exécute MQIPT. Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

MaxConnectionThreads

Le nombre maximal d'unités d'exécution de connexion, et de fait, le nombre maximal de connexions simultanées pouvant être traitées par cette route. Si cette limite est atteinte, la valeur de **MaxConnectionThreads** indique aussi le nombre des connexions mises en file d'attente lorsque toutes les unités d'exécution sont en cours d'utilisation. Lorsque ce nombre est dépassé, les requêtes de connexion suivantes sont refusées.

La valeur minimale autorisée est la valeur la plus élevée des deux valeurs suivantes : soit 1, soit la valeur de **MinConnectionThreads**.

Si la valeur est augmentée, la nouvelle valeur est utilisée lorsque la commande d'actualisation est émise. Toutes les connexions utilisent immédiatement la nouvelle valeur. La route n'est pas arrêtée.

Si la valeur est diminuée, la nouvelle valeur prend effet uniquement lorsque la route est redémarrée.

MinConnectionThreads

Le nombre d'unités d'exécution de connexions allouées pour gérer les connexions entrantes sur une route, une fois celle-ci démarrée. Le nombre d'unités d'exécution allouées ne tombe pas en-dessous de cette valeur tant que la route est active.

La valeur doit être comprise entre 0 et la valeur de **MaxConnectionThreads**.

Les modifications apportées à cette propriété prennent effet uniquement lorsque la route est redémarrée.

Nom

Un nom identifiant la route. Cette propriété est facultative. La valeur s'affiche dans les messages de la console et les informations de traçage. Les modifications apportées à cette propriété prennent effet uniquement lorsque la route est redémarrée.

OutgoingPort

Le numéro de port de départ utilisé par les connexions sortantes. La plage de numéros de port correspond à la valeur **MaxConnectionThread** de cette route. La valeur par défaut de 0 utilise un numéro de port défini par le système. Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées. Lorsque HTTP est utilisé, chaque connexion de canal requiert deux ports sortants. Pour plus d'informations, voir [Contrôle du numéro de port](#).

V 9.4.0

PasswordProtection

Indique si MQIPT peut ajouter ou supprimer la protection des données d'identification envoyées dans les structures MQCSP par IBM MQ clients, afin de maintenir la compatibilité entre le client et le gestionnaire de files d'attente, pour les routes MQIPT configurées pour ajouter ou supprimer le chiffrement TLS.

Les données d'identification dans les structures MQCSP peuvent être protégées à l'aide de la fonction de protection par mot de passe IBM MQ MQCSP ou chiffrées à l'aide du chiffrement TLS. La protection par mot de passe MQCSP est utile à des fins de test et de développement car elle est plus simple que la configuration du chiffrement TLS, mais elle n'est pas aussi sécurisée.

Pour plus d'informations sur la protection par mot de passe MQCSP, voir [Protection par mot de passe MQCSP](#).

Lorsqu'une route MQIPT est configurée pour ajouter ou supprimer le chiffrement TLS, MQIPT peut avoir besoin de protéger les données d'identification dans la structure MQCSP ou de supprimer la protection par mot de passe MQCSP pour que la connexion aboutisse.

La valeur de la propriété peut être l'une des valeurs suivantes:

obligatoire

MQIPT garantit que les données d'identification de la structure MQCSP sont chiffrées à l'aide de TLS ou protégées par un mot de passe MQCSP.

Si les données d'identification de la structure MQCSP sont envoyées chiffrées par le client à l'aide du chiffrement TLS et que la route MQIPT supprime le chiffrement TLS, MQIPT protège les données d'identification avec la protection par mot de passe MQCSP avant de transférer les données d'identification à la destination de route. Cela se produit lorsque la route MQIPT est configurée avec `SSLServer=true` et `SSLClient=false` et que la CipherSuite sélectionnée n'utilise pas de chiffrement null.

Si les données d'identification de la structure MQCSP sont protégées par le client avec la protection par mot de passe MQCSP, MQIPT ne supprime pas la protection, même si la connexion entre MQIPT et la destination de route utilise le chiffrement TLS. Si la connexion entre MQIPT et la destination de route utilise le chiffrement TLS, la connexion peut échouer avec le code anomalie `MQRC_PASSWORD_PROTECTION_ERROR` (2594).

Il s'agit de la valeur par défaut.

compatible

MQIPT applique ou supprime la protection par mot de passe MQCSP selon les besoins pour garantir l'aboutissement de la connexion.

Si les données d'identification de la structure MQCSP sont envoyées chiffrées par le client à l'aide du chiffrement TLS et que la route MQIPT supprime le chiffrement TLS, MQIPT protège les données d'identification avec la protection par mot de passe MQCSP avant de transmettre le mot de passe à la destination de la route. Cela se produit lorsque la route MQIPT est configurée avec `SSLServer=true` et `SSLClient=false` et que la CipherSuite sélectionnée n'utilise pas de chiffrement null.

Si les données d'identification de la structure MQCSP sont protégées par le client avec la protection par mot de passe MQCSP et que la route MQIPT ajoute le chiffrement TLS, MQIPT supprime la protection par mot de passe MQCSP avant de transmettre les données d'identification à la destination de la route. Cela se produit lorsque la route MQIPT est configurée avec `SSLServer=false` et `SSLClient=true` et que la CipherSuite sélectionnée n'utilise pas de chiffrement null.

Cette option offre la meilleure compatibilité. Toutefois, il ne doit être utilisé qu'à des fins de test et de développement sur des réseaux de confiance, car il ne garantit pas que le mot de passe est protégé sur le réseau.

PASSTHRU

Les données d'identification de la structure MQCSP sont transmises à la destination de route par MQIPT sans ajout ni suppression de la protection par mot de passe MQCSP. Si la route MQIPT est configurée pour ajouter ou supprimer le chiffrement TLS, les connexions client risquent d'échouer avec le code anomalie `MQRC_PASSWORD_PROTECTION_ERROR` (2594).

QMgrAccess

Définissez **QMgrAccess** sur `true` pour autoriser les connexions entrantes du canal du gestionnaire de files d'attente entrante (par exemple des canaux expéditeurs). Si vous définissez cette propriété sur `false`, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

RouteRestart

Définissez **RouteRestart** sur `false` pour empêcher le redémarrage de la route lorsque d'autres propriétés de route ont été modifiées et qu'une commande d'actualisation a été émise. La valeur par défaut de cette propriété est `true`.

SecurityExit

Définissez **SecurityExit** sur `true` pour activer un exit de sécurité défini par l'utilisateur. La valeur par défaut de cette propriété est `false`.

SecurityExitName

Nom de classe de l'exit de sécurité défini par l'utilisateur. Cette propriété doit être définie si **SecurityExit** a été défini sur `true`. Si vous modifiez cette propriété (et si **SecurityExit** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SecurityExitPath

Nom de chemin complet contenant l'exit de sécurité défini par l'utilisateur. Si cette propriété n'a pas été définie, sa valeur par défaut est le sous-répertoire des exits. Cette propriété peut également définir le nom d'un fichier d'archive Java (JAR) contenant l'exit de sécurité défini par l'utilisateur. Si vous modifiez cette propriété (et si **SecurityExit** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SecurityExitTimeout

La valeur de délai d'attente (en secondes) utilisée par MQIPT pour déterminer combien de temps il doit attendre une réponse lors de la validation d'une demande de connexion. La valeur par défaut est 30. Si vous modifiez cette propriété (et si **SecurityExit** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SocksClient

Définissez **SocksClient** sur `true` pour que la route se comporte comme un client SOCKS, puis définissez toutes les connexions via le proxy SOCKS avec les propriétés **SocksProxyHost** et **SocksProxyPort**. Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées. Cette propriété ne peut pas être utilisée avec :

- **HTTP**
- **SocksServer**
- **SSLClient**
- **SSLProxyMode**

SocksProxyHost

Nom d'hôte (ou adresse IPv4 en notation décimale à point) du proxy SOCKS utilisé par toutes les connexions pour cette route. Si vous modifiez cette propriété (et si **SocksClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées. Lorsque la propriété **SocksProxyHost** est utilisée, la propriété **Destination** doit utiliser le format de notation décimale à points.

SocksProxyPort

Le numéro du port à utiliser sur le proxy SOCKS. La valeur par défaut est 1080. Si vous modifiez cette propriété (et si **SocksClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SocksServer

Définissez **SocksServer** sur `true` pour que la route se comporte comme un proxy SOCKS et accepte les connexions client SOCKS. Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées. Cette propriété ne peut pas être utilisée avec :

- **SocksClient**
- **SSLProxyMode**
- **SSLServer**

SSLClient

Associez le paramètre **SSLClient** à la valeur `true` pour que la route se comporte comme un client SSL/TLS et établissent des connexions SSL/TLS sortantes. Lorsque vous associez le paramètre

SSLClient à la valeur `true`, la destination doit être une autre instance de MQIPT se comportant comme un serveur SSL/TLS ou un proxy/serveur HTTP.

Si vous affectez à **SSLClient** la valeur `true`, vous devez spécifier un fichier de clés client SSL/TLS à l'aide de la propriété **SSLClientKeyRing** ou **SSLClientCAKeyRing**, ou configurer MQIPT pour utiliser du matériel de cryptographie en définissant la propriété **SSLClientKeyRingUseCryptoHardware** ou **SSLClientCAKeyRingUseCryptoHardware**.

Si vous modifiez **SSLClient**, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

Cette propriété ne peut pas être utilisée avec :

- **SSLProxyMode**

SSLClientCAKeyRing

Nom de fichier qualifié complet du fichier de clés contenant les certificats de l'autorité de certification, utilisé pour authentifier les certificats du serveur SSL/TLS. Sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\) comme séparateur de fichier. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientCAKeyRingPW

Mot de passe permettant d'ouvrir le fichier de clés de l'autorité de certification du client SSL/TLS spécifié avec la propriété **SSLClientCAKeyRing** ou de se connecter au magasin de clés du matériel de cryptographie si la propriété **SSLClientCAKeyRingUseCryptoHardware** est définie sur `true`.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande `mqiptPW`, soit le nom de fichier qualifié complet du fichier contenant un mot de passe chiffré. Si vous spécifiez un nom de fichier sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\) comme séparateur de fichier. Vous êtes invité à migrer tous les mots de passe de fichier de clés actuellement stockés dans un fichier afin d'utiliser la méthode de protection la plus récente et la plus sécurisée, en rechiffant les mots de passe à l'aide de l'utilitaire `mqiptPW`. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir [Chiffrement des mots de passe stockés](#).

Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientCAKeyRingUseCryptoHardware

Indique si le matériel de cryptographie qui prend en charge l'interface PKCS #11 est utilisé comme magasin de clés pour les certificats de l'autorité de certification utilisés pour authentifier les certificats du serveur SSL/TLS, lorsque MQIPT agit en tant que client SSL/TLS. Si cette propriété est définie sur `true`, **SSLClientCAKeyRing** ne peut pas être défini sur la même route.

Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

L'utilisation du matériel de cryptographie avec MQIPT est une fonction IBM MQ Advanced. Pour utiliser cette fonction, le gestionnaire de files d'attente local qui est connecté à l'aide de la route MQIPT doit également disposer des droits IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS ou IBM MQ Advanced for z/OS VUE. La route ne démarre pas lorsque cette propriété est définie sur `true`, sauf si la propriété globale **EnableAdvancedCapabilities** est définie pour confirmer que les fonctions IBM MQ Advanced peuvent être utilisées.

SSLClientCipherSuites

Nom de la suite de chiffrement SSL/TLS à utiliser côté client SSL/TLS. Il peut s'agir d'un ou de plusieurs des suites de chiffrement prises en charge. Si vous laissez cette propriété vide, tout CipherSuite pour les protocoles activés qui est compatible avec le certificat client dans le fichier de clés est utilisé. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route

est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLClientConnectTimeout

Durée (en secondes) pendant laquelle un client SSL/TLS attend qu'une connexion SSL/TLS soit acceptée. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientCustomOutboundSNI

Indique la valeur de l'indication de nom de serveur (SNI) lorsque MQIPT initie une connexion TLS à la destination de route, si la route est configurée avec **SSLClientOutboundSNI** défini sur `custom`. Utilisez cette propriété pour définir le SNI sur une valeur spécifique qui ne peut pas être définie automatiquement par MQIPT. Par exemple, si vous souhaitez définir le SNI sur un nom d'hôte, mais que la destination de route est configurée avec une adresse IP.

La valeur doit être un nom de domaine internationalisé (IDN) valide conforme à la spécification RFC 3490 et ne peut pas se terminer par un point de fin. La route ne démarre pas si une valeur non valide est spécifiée.

Si vous modifiez la valeur de cette propriété et que **SSLClientOutboundSNI** est défini sur `custom`, la route est arrêtée et redémarrée lorsqu'une commande d'actualisation est émise.



Avertissement : Vous ne devez pas utiliser ce paramètre lorsque vous transmettez des connexions à un canal IBM MQ dont le libellé de certificat est configuré dans la zone **CERTLABL** du canal. Si vous réacheminez un client de cette manière, il sera rejeté avec un code retour MQRC_SSL_INITIALIZATION_ERROR et une erreur AMQ9673 imprimée dans les journaux d'erreurs du gestionnaire de files d'attente éloignées.

SSLClientDN_C

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce nom de pays. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les noms de pays sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_CN

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce nom usuel. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les noms usuels sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_DC

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce composant de domaine. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Vous pouvez spécifier plusieurs composants de domaine (DC) en les séparant par des virgules. Chaque composant de domaine représente un élément dans un nom de domaine, par exemple le nom de domaine `example.ibm.com` est représenté comme `example,ibm,com`. Les virgules séparent chacune des valeurs. Si vous ne spécifiez pas cette propriété, tous les composants de domaine sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_DNQ

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce qualificateur de domaine. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les qualificants de domaine sont acceptés. Si vous modifiez cette propriété

(et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_L

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à cet emplacement. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, vous désignez implicitement "tous les emplacements". Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_O

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à cette organisation. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats de toutes les organisations sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_OU

Utilisez cette propriété pour accepter des certificats reçus du client SSL/TLS correspondant à l'unité organisationnelle (OU). Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. Vous pouvez spécifier plusieurs unités organisationnelles en les séparant par des virgules. Pour faire correspondre une virgule littérale, préfixez-la avec une barre oblique inversée (\). La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'unité organisationnelle. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLClientDN_PC

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce code postal. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les codes postaux sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_ST

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à cet état. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats de tous les serveurs quel que soit leur état sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_Street

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce nom de rue. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les noms de rue sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_T

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à ce titre. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les titres sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientDN_UID

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS correspondant à cet ID utilisateur. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, tous les ID sont acceptés. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientExit

Utilisez cette propriété pour activer ou désactiver l'utilisation d'un exit lorsque la route agit en tant que client SSL/TLS. Ainsi, vous pouvez définir les caractéristiques de l'exit dans le fichier de configuration sans qu'elles soient réellement utilisées.

SSLClientKeyRing

Nom de fichier qualifié complet du fichier de clés contenant le certificat client. Sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\\) comme séparateur de fichier. Si vous modifiez **SSLClientKeyRing** (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientKeyRingPW

Mot de passe permettant d'ouvrir le fichier de clés du client SSL/TLS spécifié avec la propriété **SSLClientKeyRing** ou de se connecter au magasin de clés du matériel de cryptographie si la propriété **SSLClientKeyRingUseCryptoHardware** est définie sur `true`.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande **mqiPTW**, soit le nom de fichier qualifié complet du fichier contenant un mot de passe chiffré. Si vous spécifiez un nom de fichier sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\\) comme séparateur de fichier. Vous êtes invité à migrer tous les mots de passe de fichier de clés actuellement stockés dans un fichier afin d'utiliser la méthode de protection la plus récente et la plus sécurisée, en rechiffant les mots de passe à l'aide de l'utilitaire **mqiPTW**. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir [Chiffrement des mots de passe stockés](#).

Si vous modifiez **SSLClientKeyRingPW** (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientKeyRingUseCryptoHardware

Indique si le matériel de cryptographie qui prend en charge l'interface PKCS #11 est utilisé comme magasin de clés contenant le certificat client, lorsque MQIPT agit en tant que client SSL/TLS. Si cette propriété est définie sur `true`, **SSLClientKeyRing** ne peut pas être défini sur la même route.

Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

L'utilisation du matériel de cryptographie avec MQIPT est une fonction IBM MQ Advanced. Pour utiliser cette fonction, le gestionnaire de files d'attente local qui est connecté à l'aide de la route MQIPT doit également disposer des droits IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS ou IBM MQ Advanced for z/OS VUE. La route ne démarre pas lorsque cette propriété est définie sur `true`, sauf si la propriété globale **EnableAdvancedCapabilities** est définie pour confirmer que les fonctions IBM MQ Advanced peuvent être utilisées.

SNI SSLClientOutbound

Indique la valeur de l'extension [Server Name Indication](#) (SNI) lorsque MQIPT lance une connexion TLS à la destination de route. L'interface SNI est utilisée par les gestionnaires de files d'attente IBM MQ pour présenter le certificat correct lors de l'établissement de liaison TLS ou pour acheminer les connexions vers la destination, en fonction de la configuration.

Cette propriété s'applique uniquement aux routes définies avec `SSLClient=true` et ne peut pas être spécifiée pour les routes définies avec `HTTP=true`. Si vous modifiez la valeur de cette propriété

et que **SSLClient** est défini sur `true`, la route est arrêtée et redémarrée lorsqu'une commande d'actualisation est émise.



Avertissement : Si le canal de destination est configuré avec un libellé de certificat dans la zone **CERTLABL** de l'objet canal, vous devez définir le paramètre **CERTLABL** sur la valeur de canal. Si un client est réacheminé sans le paramètre SNI du canal, il est rejeté avec un code retour `MQRC_SSL_INITIALIZATION_ERROR` et un message `AMQ9673` imprimé dans les journaux des erreurs du gestionnaire de files d'attente éloignées.

La valeur de la propriété peut être l'une des valeurs suivantes:

nom d'hôte

Le SNI est défini sur le nom d'hôte de la destination de route. Utilisez cette option si la route se connecte à un équilibreur de charge ou à un routeur qui utilise l'interface SNI pour acheminer les demandes. Par exemple, Red Hat® OpenShift® Container Platform Router utilise l'interface SNI pour acheminer les demandes vers le gestionnaire de files d'attente IBM MQ .

Si la destination de route est un gestionnaire de files d'attente, les demandes de connexion reçoivent le certificat par défaut du gestionnaire de files d'attente éloignées lors de l'établissement de liaison TLS, et les certificats par canal ne peuvent donc pas être utilisés.

Si la destination de route est spécifiée à l'aide d'une adresse IP et qu'une recherche DNS inversée ne peut pas être effectuée, la valeur SNI est vide.

Il s'agit de la valeur par défaut.

canal

Le SNI est défini sur le nom de canal IBM MQ . Utilisez cette option pour autoriser les certificats par canal à être utilisés par le gestionnaire de files d'attente de destination si les connexions reçues par la route ne contiennent pas le nom de canal dans le SNI pour l'une des raisons suivantes:

- La route est configurée pour accepter les connexions qui ne sont pas sécurisées avec TLS avec `SSLServer=false` ou `SSLPlainConnections=true`.
- L'application qui se connecte à la route ne peut pas définir le SNI ou est configurée pour définir le SNI sur une valeur autre que le nom de canal IBM MQ .

PASSTHRU

Si la route est définie avec `SSLServer=true`, le SNI sur la connexion sortante est défini sur la valeur du SNI reçu sur la connexion entrante à la route. Si la route n'est pas configurée pour accepter les connexions TLS, l'interface SNI est définie sur le nom d'hôte de destination.

personnalisée

La valeur SNI est définie sur la valeur spécifiée dans la propriété

SSLClientCustomOutboundSNI . Si la propriété **SSLClientCustomOutboundSNI** n'est pas spécifiée, le SNI est défini comme si la route était configurée avec `SSLClientOutboundSNI=hostname`.

aucun

Le SNI n'est pas défini.

SSLClientProtocols

Permet de restreindre l'ensemble des protocoles de connexions sécurisées utilisés pour établir des connexions sortantes vers la destination pour une route, lorsque **SSLClient** est défini sur `true`.

Vous pouvez spécifier plusieurs valeurs en les séparant par des virgules. Si vous ne spécifiez pas cette propriété, TLS 1.2 et TLS 1.3 sont activés par défaut. Pour activer des protocoles autres que TLS 1.2 ou TLS 1.3, vous devez spécifier les protocoles à activer dans cette propriété et ajouter la prise en charge du protocole dans Java runtime environment en suivant la procédure décrite dans [Activation des protocoles obsolètes et des suites de chiffrement CipherSuites](#). Vous pouvez spécifier une ou plusieurs des valeurs suivantes.

<i>Tableau 118. Valeurs admises pour les protocoles SSL/TLS</i>	
Valeur	Protocole
SSLv3	SSL 3.0
TLSv1	TLS 1.0
TLSv1.1	TLS 1.1
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

Utilisez l'entrée listée dans la colonne **Valeur** de la propriété de route. L'entrée correspondante dans la colonne **Protocole** est fournie à titre d'information seulement.

SSLClientSiteDN_C

Utilisez cette propriété pour spécifier un nom de pays afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de pays. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_CN

Utilisez cette propriété pour spécifier un nom usuel afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom usuel. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_DC

Utilisez cette propriété pour spécifier un nom de composant de domaine afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Vous pouvez spécifier plusieurs composants de domaine (DC) en les séparant par des virgules. Chaque composant de domaine représente un élément dans un nom de domaine, par exemple le nom de domaine `example.ibm.com` est représenté comme `example,ibm,com`. Les virgules séparent chacune des valeurs. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de composant de domaine. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_DNQ

Utilisez cette propriété pour spécifier un qualificateur de domaine afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel qualifiant de composant de domaine. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_L

Utilisez cette propriété pour spécifier un nom d'emplacement afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'emplacement. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_O

Utilisez cette propriété pour spécifier un nom d'organisation afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'organisation. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est

arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_OU

Utilisez cette propriété pour spécifier un nom d'unité organisationnelle (OU) afin de sélectionner un certificat à envoyer au serveur SSL/TLS. Vous pouvez spécifier plusieurs unités organisationnelles en les séparant par des virgules. Pour faire correspondre une virgule littérale, préfixez-la avec une barre oblique inversée (\). La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'unité organisationnelle. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLClientSiteDN_PC

Utilisez cette propriété pour spécifier un code postal afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel code postal. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_ST

Utilisez cette propriété pour spécifier un nom d'état afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel code d'état. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_Street

Utilisez cette propriété pour spécifier un nom de rue afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de rue. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_T

Utilisez cette propriété pour spécifier un titre afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel titre. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteDN_UID

Utilisez cette propriété pour spécifier un ID utilisateur afin de sélectionner un certificat à envoyer au serveur SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel ID utilisateur. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLClientSiteLabel

Utilisez cette propriété pour spécifier un nom de label afin de sélectionner un certificat à envoyer au serveur SSL/TLS. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de label. Si vous modifiez cette propriété (et si **SSLClient** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLExitData

Utilisez cette propriété pour fournir une chaîne définie par l'utilisateur à transmettre à l'exit.

SSLExitName

Utilisez cette propriété pour définir le nom de classe de l'exit qui doit être appelé lorsque la route agit en tant que client ou serveur SSL/TLS. Le nom doit inclure un nom du pack, par exemple, `com.ibm.mq.ipc.exit.TestExit`.

SSLExitPath

Cette propriété définit l'emplacement de l'exit à utiliser pour charger une copie de l'exit. Le nom doit être un nom qualifié complet permettant de rechercher le fichier de classe ou le nom d'un fichier .jar contenant le fichier de classe : par exemple, C:\mqipt\exits ou C:\mqipt\exits\exits.jar.

SSLExitTimeout

Cette propriété définit la durée pendant laquelle MQIPT attend la fin d'exécution de l'exit avant d'arrêter la demande de connexion. La valeur 0 signifie que MQIPT attend indéfiniment.

SSLPlainConnections

Utilisez cette propriété pour indiquer si SSL/TLS est obligatoire pour les connexions au port d'écoute de MQIPT d'une route configurée pour accepter les connexions SSL/TLS entrantes. Cette propriété s'applique aux routes pour lesquelles la propriété **SSLServer** ou **SSLProxyMode** a pour valeur true. Si elle est activée, cette propriété autorise les connexions non chiffrées au port d'écoute de la route, c'est-à-dire que MQIPT peut transmettre toutes les connexions IBM MQ au port d'écoute du gestionnaire de files d'attente, que la connexion soit ou non chiffrée. Si vous ne définissez pas ce paramètre ou si vous l'associez à la valeur false, seules les connexions SSL/TLS entrantes sont autorisées. Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLProxyMode

Associez cette propriété à la valeur true pour que la route accepte uniquement les demandes de connexion client SSL/TLS et tunnelle la demande directement vers la destination. Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées. Cette propriété ne peut pas être utilisée avec les propriétés suivantes :

- **SocksClient**
- **SocksServer**
- **SSLClient**
- **SSLServer**

SSLServer

Associez cette propriété à la valeur true pour que la route se comporte comme un serveur SSL/TLS et accepte les connexions SSL/TLS entrantes. Si vous associez **SSLServer** à la valeur true, l'appelant doit être une autre instance de MQIPT agissant comme un client SSL/TLS, ou un client ou gestionnaire de files d'attente IBM MQ pour lequel le protocole SSL/TLS est activé.

Si vous définissez **SSLServer** sur true, vous devez spécifier un fichier de clés de serveur SSL/TLS à l'aide de la propriété **SSLServerKeyRing** ou configurer MQIPT pour qu'il utilise du matériel de cryptographie en définissant la propriété **SSLServerKeyRingUseCryptoHardware**.

Si vous modifiez cette propriété, la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

Cette propriété ne peut pas être utilisée avec les propriétés suivantes :

- **SocksServer**
- **SSLProxyMode**

SSLServerCAKeyRing

Nom de fichier qualifié complet du fichier de clés contenant les certificats de l'autorité de certification, utilisé pour authentifier les certificats du client SSL/TLS. Sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\) comme séparateur de fichier. Si vous modifiez cette propriété (et si **SSLServer** est défini sur true), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLServerCAKeyRingPW

Mot de passe permettant d'ouvrir le fichier de clés de l'autorité de certification du serveur SSL/TLS spécifié avec la propriété **SSLServerCAKeyRing** ou de se connecter au magasin de clés du matériel de cryptographie si la propriété **SSLServerCAKeyRingUseCryptoHardware** est définie sur true.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande **mqiPTW**, soit le nom de fichier qualifié complet du fichier contenant un mot de passe chiffré. Si vous spécifiez un nom de fichier sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\\) comme séparateur de fichier. Vous êtes invité à migrer tous les mots de passe de fichier de clés actuellement stockés dans un fichier afin d'utiliser la méthode de protection la plus récente et la plus sécurisée, en rechiffrant les mots de passe à l'aide de l'utilitaire **mqiPTW**. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir [Chiffrement des mots de passe stockés](#).

Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerCAKeyRingUseCryptoHardware

Indique si le matériel de cryptographie qui prend en charge l'interface PKCS #11 est utilisé comme magasin de clés pour les certificats de l'autorité de certification, utilisés pour authentifier les certificats du client SSL/TLS. Si cette propriété est définie sur `true`, **SSLServerCAKeyRing** ne peut pas être défini sur la même route.

Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

L'utilisation du matériel de cryptographie avec MQIPT est une fonction IBM MQ Advanced. Pour utiliser cette fonction, le gestionnaire de files d'attente local qui est connecté à l'aide de la route MQIPT doit également disposer des droits IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS ou IBM MQ Advanced for z/OS VUE. La route ne démarre pas lorsque cette propriété est définie sur `true`, sauf si la propriété globale **EnableAdvancedCapabilities** est définie pour confirmer que les fonctions IBM MQ Advanced peuvent être utilisées.

SSLServerAskClientAuth

Utilisez cette propriété pour demander au serveur SSL/TLS d'authentifier le client SSL/TLS. Le client SSL/TLS doit posséder son propre certificat à envoyer au serveur SSL/TLS. Le certificat est extrait du fichier de clés. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLServerCipherSuites

Nom de la suite de chiffrement SSL/TLS à utiliser côté serveur SSL/TLS. Il peut s'agir d'un ou de plusieurs des suites de chiffrement prises en charge. Si vous laissez cette zone vide, toute CipherSuite pour les protocoles activés qui est compatible avec le certificat serveur dans le fichier de clés est utilisée. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLServerDN_C

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce nom de pays. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de société. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_CN

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce nom usuel. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom usuel. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_DC

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce nom de composant de domaine. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Vous pouvez spécifier plusieurs composants de domaine (DC) en les séparant par des virgules. Chaque composant de domaine représente un élément dans un nom de domaine, par exemple le nom de domaine `example.ibm.com` est représenté comme `example,ibm,com`. Les virgules séparent chacune des valeurs. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de composant de domaine. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_DNQ

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce qualificateur de domaine. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel qualifiant de composant de domaine. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_L

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de cet emplacement. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel emplacement. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_O

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de cette organisation. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quelle organisation. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_OU

Utilisez cette propriété pour accepter des certificats reçus du client SSL/TLS de cette unité organisationnelle (OU). Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. Vous pouvez spécifier plusieurs unités organisationnelles en les séparant par des virgules. Pour faire correspondre une virgule littérale, préfixez-la avec une barre oblique inversée (\). La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'unité organisationnelle. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLServerDN_PC

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce code postal. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel code postal. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_ST

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de cet état. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel état. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_Street

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce nom de rue. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de rue. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_T

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de ce titre. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel titre. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerDN_UID

Utilisez cette propriété pour accepter des certificats reçus du serveur SSL/TLS de cet ID utilisateur. Le nom peut comporter un astérisque (*) en préfixe ou suffixe pour étendre sa portée. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel ID utilisateur. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerExit

Utilisez cette propriété pour activer ou désactiver l'utilisation d'un exit lorsque la route agit en tant que serveur SSL/TLS. Ainsi, vous pouvez définir les caractéristiques de l'exit dans le fichier de configuration sans qu'elles soient réellement utilisées.

SSLServerKeyRing

Nom de fichier qualifié complet du fichier de clés contenant le certificat serveur. Sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\\) comme séparateur de fichier. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerKeyRingPW

Mot de passe permettant d'ouvrir le fichier de clés du serveur SSL/TLS spécifié avec la propriété **SSLServerKeyRing** ou de se connecter au magasin de clés du matériel de cryptographie si la propriété **SSLServerKeyRingUseCryptoHardware** est définie sur `true`.

La valeur peut être soit un mot de passe chiffré à l'aide de la commande **mqiPTW**, soit le nom de fichier qualifié complet du fichier contenant un mot de passe chiffré. Si vous spécifiez un nom de fichier sur les plateformes Windows, vous devez utiliser une double barre oblique inversée (\\) comme séparateur de fichier. Vous êtes invité à migrer tous les mots de passe de fichier de clés actuellement stockés dans un fichier afin d'utiliser la méthode de protection la plus récente et la plus sécurisée, en rechiffrant les mots de passe à l'aide de l'utilitaire **mqiPTW**. Pour plus d'informations sur le chiffrement des mots de passe dans la configuration MQIPT, voir [Chiffrement des mots de passe stockés](#).

Vous devez spécifier **SSLServerKeyRingPW** si vous définissez **SSLServer** sur `true`.

Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerKeyRingUseCryptoHardware

Indique si le matériel de cryptographie qui prend en charge l'interface PKCS #11 est utilisé comme magasin de clés pour le certificat serveur, lorsque MQIPT agit en tant que serveur SSL/TLS. Si cette propriété est définie sur `true`, **SSLServerKeyRing** ne peut pas être défini sur la même route.

Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

L'utilisation du matériel de cryptographie avec MQIPT est une fonction IBM MQ Advanced . Pour utiliser cette fonction, le gestionnaire de files d'attente local qui est connecté à l'aide de la route MQIPT doit également disposer des droits IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS ou IBM MQ Advanced for z/OS VUE . La route ne démarre pas lorsque cette propriété est définie sur `true` , sauf si la propriété globale **EnableAdvancedCapabilities** est définie pour confirmer que les fonctions IBM MQ Advanced peuvent être utilisées.

SSLServerProtocols

Permet de restreindre l'ensemble des protocoles de connexion sécurisée utilisés pour accepter des connexions sortantes vers le port d'écoute d'une route lorsque **SSLServer** a pour valeur `true`.

Vous pouvez spécifier plusieurs valeurs en les séparant par des virgules. Si vous ne spécifiez pas cette propriété, TLS 1.2 et TLS 1.3 sont activés par défaut. Pour activer des protocoles autres que TLS 1.2 ou TLS 1.3, vous devez spécifier les protocoles à activer dans cette propriété et ajouter la prise en charge du protocole dans Java runtime environment en suivant la procédure décrite dans [Activation des protocoles obsolètes et des suites de chiffrement CipherSuites](#). Vous pouvez spécifier une ou plusieurs des valeurs suivantes.

Valeur	Protocole
SSLv3	SSL 3.0
TLSv1	TLS 1.0
TLSv1.1	TLS 1.1
TLSv1.2	TLS 1.2
TLSv1.3	TLS 1.3

Utilisez l'entrée listée dans la colonne **Valeur** de la propriété de route. L'entrée correspondante dans la colonne **Protocole** est fournie à titre d'information seulement.

SSLServerSiteDN_C

Utilisez cette propriété pour spécifier un nom de pays afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de pays. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_CN

Utilisez cette propriété pour spécifier un nom usuel afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom usuel. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_DC

Utilisez cette propriété pour spécifier un nom de composant de domaine afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Vous pouvez spécifier plusieurs composants de domaine (DC) en les séparant par des virgules. Chaque composant de domaine représente un élément dans un nom de domaine, par exemple le nom de domaine `example.ibm.com` est représenté comme `example,ibm,com`. Les virgules séparent chacune des valeurs. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de composant de domaine. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_DNQ

Utilisez cette propriété pour spécifier un qualificateur de domaine afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel qualifiant de composant de domaine. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_L

Utilisez cette propriété pour spécifier un nom d'emplacement afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'emplacement. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_O

Utilisez cette propriété pour spécifier un nom d'organisation afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'organisation. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_OU

Utilisez cette propriété pour spécifier un nom d'unité organisationnelle (OU) afin de sélectionner un certificat à envoyer au client SSL/TLS. Vous pouvez spécifier plusieurs unités organisationnelles en les séparant par des virgules. Pour faire correspondre une virgule littérale, préfixez-la avec une barre oblique inversée (\). La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom d'unité organisationnelle. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à cette route sont arrêtées.

SSLServerSiteDN_PC

Utilisez cette propriété pour spécifier un code postal afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel code postal. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_ST

Utilisez cette propriété pour spécifier un nom d'état afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel code d'état. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_Street

Utilisez cette propriété pour spécifier un nom de rue afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de rue. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_T

Utilisez cette propriété pour spécifier un titre afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel titre. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteDN_UID

Utilisez cette propriété pour spécifier un ID utilisateur afin de sélectionner un certificat à envoyer au client SSL/TLS. La mise en correspondance des certificats n'est pas sensible à la casse. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel ID utilisateur. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

SSLServerSiteLabel

Utilisez cette propriété pour spécifier un nom de label afin de sélectionner un certificat à envoyer au client SSL/TLS. Si vous ne spécifiez pas cette propriété, les certificats sont acceptés avec n'importe quel nom de label. Si vous modifiez cette propriété (et si **SSLServer** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise. Toutes les connexions à la route sont arrêtées.

Format StoredCredentials

Utilisez cette propriété pour indiquer si les valeurs des propriétés de mot de passe utilisent le format de mot de passe chiffré pris en charge dans MQIPT à partir de IBM MQ 9.1.5. MQIPT peut presque toujours détecter si les mots de passe sont spécifiés au format de mot de passe chiffré. Cette propriété doit uniquement être définie dans le scénario improbable où MQIPT ne peut pas différencier automatiquement un mot de passe chiffré d'un mot de passe en texte en clair ou d'un nom de fichier.

Les valeurs admises sont les suivantes :

chiffré

Les propriétés de mot de passe contiennent un mot de passe chiffré au format pris en charge dans MQIPT à partir de IBM MQ 9.1.5.

COMPAT

Les propriétés de mot de passe contiennent un mot de passe en texte en clair ou, pour les mots de passe de fichier de clés, le nom du fichier contenant un mot de passe chiffré.

TCPKeepAlive

Définissez cette propriété sur `true` pour activer l'envoi régulier de paquets TCP/IP de signal de présence servant à maintenir actives les connexions sur cette route. Cette opération réduit le risque que les connexions MQIPT soient annulées par un pare-feu ou un routeur. L'envoi des paquets TCP/IP de signal de présence est régi par les paramètres d'optimisation du système d'exploitation. Consultez la documentation de votre système d'exploitation pour plus d'informations pour savoir comment optimiser le signal de présence. Si vous ne définissez pas ce paramètre, ou le définissez sur `false`, les paquets de signal de présence ne sont pas envoyés.

Fonction de trace

Le niveau de traçage requis pour cette route. L'activation de la trace pour une route n'active pas la trace sur les autres routes. Si vous devez tracer plusieurs routes, vous devez ajouter la propriété **Trace** à la section `[route]` de chaque route à tracer.

La valeur de cette propriété peut être l'une des suivantes:

0

La trace n'est pas activée

Tout entier positif

La trace est activée

La valeur par défaut est 0.

Si la section `[route]` n'inclut pas de propriété **Trace**, la propriété **Trace** de la section `[global]` est utilisée. Pour plus d'informations sur le traçage des unités d'exécution non associées à une route, voir **Trace** dans la section `[global]`. Si une modification de cette propriété affecte cette route, la nouvelle valeur est utilisée lorsque la commande d'actualisation est émise. Toutes les connexions utilisent immédiatement la nouvelle valeur. La route n'est pas arrêtée.

Données TraceUser

Quantité de données utilisateur dans les transmissions réseau reçues et envoyées par cette route qui est tracée, lorsque la fonction de trace est activée pour cette route. Les valeurs admises sont les suivantes :

0

Aucune donnée utilisateur n'est tracée.

tout

Toutes les données utilisateur sont tracées.

numberOfoctets

Le nombre spécifié d'octets de données, y compris l'en-tête de segment de transmission (TSH), est tracé. La valeur indiquée doit être supérieure à 15.

UriName

Cette propriété peut être utilisée pour modifier le nom de l'identificateur URI (Uniform Resource Identifier) de la ressource lors de l'utilisation d'un proxy HTTP, bien que la valeur par défaut suffise pour la plupart des configurations :

```
HTTP://destination:destination_port/mqipt
```

Si vous modifiez cette propriété (et si **HTTP** est défini sur `true`), la route est arrêtée, puis redémarrée lorsqu'une commande d'actualisation est émise.

mqiptAdminpropriétés

La commande **mqiptAdmin** lit les propriétés de configuration à partir d'un fichier de propriétés spécifié lors du démarrage de la commande.

Les propriétés suivantes peuvent être spécifiées dans le fichier de propriétés utilisé par la commande **mqiptAdmin**. Les noms de propriété sont sensibles à la casse.

PasswordProtectionKeyFile

Nom du fichier contenant la clé de chiffrement utilisée pour chiffrer le mot de passe du magasin de clés de confiance spécifié dans la propriété **SSLClientCAKeyRingPW**. Si cette propriété n'est pas spécifiée, la clé de chiffrement par défaut est utilisée pour déchiffrer le mot de passe. La clé de chiffrement utilisée pour chiffrer le mot de passe du magasin de clés de confiance **mqiptAdmin** peut être différente de la clé de chiffrement utilisée pour chiffrer les mots de passe dans le fichier de configuration `mqipt.conf`.

SSLClientCAKeyRing

Nom de fichier du magasin de clés de confiance PKCS#12 à utiliser pour les connexions au port de commande TLS MQIPT. Le magasin de clés de confiance doit contenir le certificat de l'autorité de certification qui a signé le certificat serveur que le port de commande TLS MQIPT est configuré pour utiliser. Les caractères de barre oblique inversée (`\`) dans le nom de fichier doivent être échappés et spécifiés sous la forme d'une double barre oblique inversée (`\\`).

SSLClientCAKeyRingPW

Mot de passe chiffré permettant d'accéder au magasin de clés de confiance spécifié à l'aide de la propriété **SSLClientCAKeyRing**. Le mot de passe doit être chiffré à l'aide de la commande **mqiptPW** et la valeur de cette propriété doit être définie sur la chaîne générée par **mqiptPW**.

Remarques

:NONE.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Consultez votre représentant IBM local pour obtenir des informations sur les produits et services actuellement disponibles dans votre région. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout produit, programme ou service fonctionnellement équivalent qui ne porte pas atteinte à un droit de propriété intellectuelle IBM peut être utilisé à la place. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pour obtenir des informations sur les licences relatives aux informations sur deux octets (DBCS), contactez le service de la propriété intellectuelle IBM de votre pays ou envoyez vos demandes de renseignements, par écrit, à :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et/ou programmes décrits dans ce document.

Les références à des sites Web non IBM sont fournies uniquement à titre d'information et n'impliquent en aucune façon une adhésion de ces sites Web. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
Coordinateur d'interopérabilité logicielle, département 49XA
3605 Autoroute 52 N

Rochester, MN 55901
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans le présent document et tous les éléments sous disponibles s'y rapportant sont fournis par IBM conformément aux dispositions du Contrat sur les produits et services IBM, aux Conditions Internationales d'Utilisation de Logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Licence sur les droits d'auteur :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Documentation sur l'interface de programmation

Les informations d'interface de programmation, si elles sont fournies, sont destinées à vous aider à créer un logiciel d'application à utiliser avec ce programme.

Ce manuel contient des informations sur les interfaces de programmation prévues qui permettent au client d'écrire des programmes pour obtenir les services d'IBM MQ.

Toutefois, lesdites informations peuvent également contenir des données de diagnostic, de modification et d'optimisation. Ces données vous permettent de déboguer votre application.

Important : N'utilisez pas ces informations de diagnostic, de modification et d'optimisation en tant qu'interface de programmation car elles sont susceptibles d'être modifiées.

Marques

IBM, le logo IBM, ibm.com, sont des marques d'IBM Corporation dans de nombreux pays. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark".

information"www.ibm.com/legal/copytrade.shtml. Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés.

Microsoft et Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque de The Open Group aux Etats-Unis et dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Ce produit inclut des logiciels développés par le projet Eclipse (<https://www.eclipse.org/>).

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses sociétés affiliées.



Référence :

(1P) P/N: