

9.0

Administration d' IBM MQ

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 499.

Cette édition s'applique à la version 9 édition 0 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 2015, 2023.**

Table des matières

| | |
|--|----------|
| Administration..... | 5 |
| Administration locale et à distance..... | 8 |
| Administration d' IBM MQ à l'aide de commandes de contrôle..... | 9 |
| Administration de MQ à l'aide des commandes MQSC..... | 10 |
| Commandes de script (MQSC)..... | 11 |
| Noms d'objet IBM MQ dans les commandes MQSC..... | 13 |
| Entrée et sortie standard..... | 14 |
| Utilisation des commandes MQSC en mode interactif..... | 14 |
| Exécution de commandes MQSC à partir de fichiers texte..... | 16 |
| Exécution de commandes MQSC à partir de fichiers de traitement par lots..... | 19 |
| Résolution des problèmes liés aux commandes MQSC..... | 20 |
| Automatisation de l'administration de IBM MQ à l'aide des commandes PCF..... | 21 |
| Introduction à IBM MQ Programmable Command Formats..... | 22 |
| Utilisation de MQAI pour simplifier l'utilisation des fichiers PCF..... | 35 |
| Administration à l'aide de REST API..... | 72 |
| Guide d'initiation à administrative REST API..... | 73 |
| Utilisation de administrative REST API..... | 79 |
| Administration à distance à l'aide de REST API..... | 85 |
| REST API Horodatages..... | 89 |
| REST API traitement des erreurs..... | 89 |
| REST API découverte..... | 92 |
| REST APISupport de langue nationale (NLS)..... | 93 |
| Administration à l'aide de IBM MQ Console..... | 95 |
| Guide d'initiation à IBM MQ Console..... | 95 |
| Utilisation des gestionnaires de files d'attente locales..... | 99 |
| Utilisation des objets IBM MQ..... | 101 |
| Utilisation d'enregistrements de droits d'accès..... | 117 |
| Surveillance de l'utilisation des ressources système..... | 120 |
| Configuration des présentations de tableau de bord..... | 132 |
| Commandes du tableau de bord..... | 134 |
| Raccourcis-clavier..... | 135 |
| Administration à l'aide de IBM MQ Explorer..... | 135 |
| Ce que vous pouvez faire avec le IBM MQ Explorer..... | 135 |
| Configuration de IBM MQ Explorer..... | 137 |
| Utilisation de l'application IBM MQ Taskbar (Windows uniquement)..... | 143 |
| L'application du moniteur d'alertes IBM MQ (Windows uniquement)..... | 144 |
| Administration des objets IBM MQ locaux..... | 144 |
| Démarrage et arrêt d'un gestionnaire de files d'attente..... | 145 |
| Arrêt des canaux MQI..... | 148 |
| Utilisation des gestionnaires de files d'attente..... | 149 |
| Utilisation des files d'attente locales..... | 150 |
| Utilisation de files d'attente alias..... | 156 |
| Utilisation des files d'attente de rebut..... | 158 |
| Utilisation des files d'attente modèles..... | 178 |
| Utilisation des rubriques d'administration..... | 179 |
| Utilisation des abonnements..... | 182 |
| Utilisation des services..... | 186 |
| Gestion des objets pour le déclenchement..... | 193 |
| Utilisation de l'utilitaire dmpmqmsg entre deux systèmes..... | 195 |
| Administration des objets IBM MQ éloignés..... | 199 |
| Canaux et mise en file d'attente à distance..... | 199 |
| Administration à distance à partir d'un gestionnaire de files d'attente local..... | 201 |

| | |
|---|------------|
| Création d'une définition locale d'une file d'attente éloignée..... | 207 |
| Vérification de la fin des commandes asynchrones pour les réseaux distribués..... | 210 |
| Utilisation de définitions de file d'attente éloignée comme alias..... | 212 |
| Conversion de données..... | 213 |
| Administration d'MQ Telemetry..... | 218 |
| Configuration d'un gestionnaire de files d'attente pour la télémétrie sous Linux et AIX..... | 218 |
| Configuration d'un gestionnaire de files d'attente pour la télémétrie sous Windows..... | 220 |
| Configuration de la mise en file d'attente répartie pour l'envoi de messages aux clients MQTT.... | 222 |
| MQTT Identification, autorisation et authentification du client..... | 225 |
| Authentification de canal de télémétrie à l'aide de TLS..... | 231 |
| Confidentialité de la publication sur les canaux de télémétrie..... | 233 |
| Configuration TLS des clients MQTT Java et des canaux de télémétrie..... | 233 |
| Configuration JAAS du canal de télémétrie..... | 238 |
| Administration d'IBM MQ Light..... | 240 |
| Affichage des objets IBM MQ utilisés par les clients MQ Light..... | 241 |
| MQ Light Identification, autorisation et authentification du client..... | 242 |
| Confidentialité des publications sur les canaux..... | 244 |
| Configuration des clients MQ Light avec TLS..... | 245 |
| Déconnexion des clients MQ Light du gestionnaire de files d'attente..... | 246 |
| Administration de la multidiffusion..... | 246 |
| Initiation à la multidiffusion..... | 246 |
| Topologie de rubrique de multidiffusion IBM MQ..... | 247 |
| Contrôle de la taille des messages de multidiffusion..... | 248 |
| Activation de la conversion de données pour la messagerie multidiffusion..... | 250 |
| Surveillance des applications de multidiffusion..... | 251 |
| Fiabilité des messages de multidiffusion..... | 252 |
| Tâches de multidiffusion avancées..... | 253 |
| Administration d'IBM MQ for IBM i..... | 256 |
| Gestion d'IBM MQ for IBM i à l'aide de commandes CL..... | 256 |
| Autres méthodes d'administration de IBM MQ for IBM i..... | 270 |
| Gestion des travaux pour IBM i..... | 276 |
| Disponibilité, sauvegarde, récupération et redémarrage sur IBM i..... | 284 |
| mise au repos IBM MQ for IBM i..... | 330 |
| Administration d'IBM MQ for z/OS..... | 334 |
| Emission de commandes sur IBM MQ for z/OS..... | 335 |
| Les utilitaires IBM MQ for z/OS..... | 344 |
| fonctionnement IBM MQ for z/OS..... | 346 |
| Ecriture de programmes pour administrer IBM MQ for z/OS..... | 368 |
| Gestion des ressources IBM MQ sur z/OS..... | 381 |
| Reprise et redémarrage sur z/OS..... | 423 |
| IBM MQ et IMS..... | 447 |
| Exploitation de Advanced Message Security sous z/OS..... | 460 |
| Fournisseur de services IBM MQ for z/OS pour z/OS Connect..... | 461 |
| Remarques..... | 499 |
| Documentation sur l'interface de programmation..... | 500 |
| Marques..... | 500 |



Administration d'IBM MQ

Pour administrer vos gestionnaires de files d'attente IBM MQ et les ressources associées, choisissez la méthode de votre choix parmi un ensemble de tâches que vous pouvez utiliser pour activer et gérer ces ressources.






Vous pouvez administrer les objets IBM MQ localement ou à distance (voir [«Administration locale et à distance»](#), à la page 8).


Il existe un certain nombre de méthodes différentes que vous pouvez utiliser pour créer et administrer vos gestionnaires de files d'attente et leurs ressources associées dans IBM MQ. Ces méthodes incluent des interfaces de ligne de commande, des interfaces utilisateur graphiques et une API d'administration.

Il existe différents ensembles de commandes que vous pouvez utiliser pour administrer IBM MQ en fonction de votre plateforme:

- [«IBM MQ Les commandes de contrôle»](#), à la page 5
- [«Commandes IBM MQ Script \(MQSC\)»](#), à la page 6
- [«Formats PCF \(Programmable Command Formats\)»](#), à la page 6
-  [administrative REST API](#)
-  [«Langage de contrôle IBM i \(CL\)»](#), à la page 7

Il existe également les autres options suivantes pour la création et la gestion des objets IBM MQ :

-   [«IBM MQ Explorer»](#), à la page 7
-  [«IBM MQ Console»](#), à la page 8
-  [«Application de configuration par défaut de Windows»](#), à la page 8
-  [«Microsoft Cluster Service \(MSCS\)»](#), à la page 8

 Pour plus d'informations sur les interfaces d'administration et les options sous IBM MQ for z/OS, voir [«Administration d'IBM MQ for z/OS»](#), à la page 334.

Vous pouvez automatiser certaines tâches d'administration et de surveillance pour les gestionnaires de files d'attente locales et éloignées à l'aide des commandes PCF. Ces commandes peuvent également être simplifiées à l'aide de l'interface d'administration IBM MQ (MQAI) sur certaines plateformes. Pour plus d'informations sur l'automatisation des tâches d'administration, voir [«Automatisation de l'administration de IBM MQ à l'aide des commandes PCF»](#), à la page 21.

IBM MQ Les commandes de contrôle



Vous pouvez effectuer des tâches d'administration sur les gestionnaires de files d'attente eux-mêmes à l'aide des commandes de contrôle.

IBM MQ for Windows, les systèmes UNIX and Linux[®] fournissent les *commandes de contrôle* que vous exécutez sur la ligne de commande du système.

Les commandes de contrôle sont décrites dans [Création et gestion de gestionnaires de files d'attente sur Multiplatforms](#). Pour obtenir les informations de référence sur les commandes de contrôle, voir [Commandes de contrôle IBM MQ](#).

Commandes IBM MQ Script (MQSC)

Les commandes MQSC permettent de gérer les objets de gestionnaire de files d'attente, y compris le gestionnaire de files d'attente lui-même, les files d'attente, les définitions de processus, les listes de noms, les canaux, les canaux de connexion client, les programmes d'écoute, les services et les objets d'informations d'authentification.

Vous émettez des commandes MQSC vers un gestionnaire de files d'attente à l'aide de la commande `runmqsc`. Vous pouvez effectuer cette opération de manière interactive, en émettant des commandes à partir d'un clavier, ou vous pouvez rediriger le périphérique d'entrée standard (stdin) pour exécuter une séquence de commandes à partir d'un fichier texte ASCII. Dans les deux cas, le format des commandes est identique.

Vous pouvez exécuter la commande `runmqsc` dans trois modes, en fonction des indicateurs définis sur la commande:

- *Mode de vérification*, dans lequel les commandes MQSC sont vérifiées sur un gestionnaire de files d'attente local, mais ne sont pas exécutées
- *Mode direct*, dans lequel les commandes MQSC sont exécutées sur un gestionnaire de files d'attente local
- *Mode indirect*, dans lequel les commandes MQSC sont exécutées sur un gestionnaire de files d'attente éloignées

Les commandes MQSC sont disponibles sur toutes les plateformes, y compris IBM i et z/OS. Les commandes MQSC sont récapitulées dans [Comparaison des ensembles de commandes](#).

ULW Sous UNIX, Linux, and Windows, vous pouvez utiliser le MQSC en tant que commandes uniques émises sur la ligne de commande du système. Pour émettre des commandes plus complexes ou multiples, le MQSC peut être intégré dans un fichier que vous exécutez à partir de la ligne de commande. Les commandes MQSC peuvent être envoyées à un gestionnaire de files d'attente éloignées. Pour plus de détails, voir [Génération de scripts de commande](#).

IBM i Pour exécuter les commandes sur un serveur IBM i, créez une liste de commandes dans un fichier script, puis exécutez le fichier à l'aide de la commande STRMQMMQSC.

Remarques : **IBM i**

1. N'utilisez pas la bibliothèque QTEMP comme bibliothèque d'entrée pour STRMQMMQSC, car l'utilisation de la bibliothèque QTEMP est limitée. Vous devez utiliser une autre bibliothèque comme fichier d'entrée de la commande.
2. Sous IBM i, les réponses MQSC aux commandes émises à partir d'un fichier script sont renvoyées dans un fichier spoule.

«[Commandes de script \(MQSC\)](#)», à la page 11 contient une description de chaque commande MQSC et de sa syntaxe.

Pour plus d'informations sur l'utilisation des commandes MQSC, voir «[Administration de MQ à l'aide des commandes MQSC](#)», à la page 10.

Formats PCF (Programmable Command Formats)

Les formats PCF (Programmable Command Formats) définissent les messages de commande et de réponse qui peuvent être échangés entre un programme et n'importe quel gestionnaire de files d'attente (qui prend en charge les PCF) d'un réseau. Vous pouvez utiliser des commandes PCF dans un programme d'application de gestion des systèmes pour l'administration des objets IBM MQ : objets d'informations d'authentification, canaux, programmes d'écoute de canal, listes de noms, définitions de processus, gestionnaires de files d'attente, files d'attente, services et classes de stockage. L'application peut fonctionner à partir d'un point unique du réseau pour communiquer des informations de commande et de réponse avec n'importe quel gestionnaire de files d'attente, local ou distant, à l'aide du gestionnaire de files d'attente local.

Pour plus d'informations sur les fichiers PCF, voir [«Introduction à IBM MQ Programmable Command Formats»](#), à la page 22.

Pour la définition des fichiers PCF et des structures des commandes et des réponses, voir [Programmable command formats reference](#).

administrative REST API

V 9.0.1

administrative REST API fournit une interface RESTful que vous pouvez utiliser pour administrer IBM MQ. Lorsque vous utilisez administrative REST API, vous appelez une méthode HTTP sur une URL qui représente un objet IBM MQ . Par exemple, vous pouvez demander des informations sur les installations IBM MQ à l'aide de la méthode HTTP GET sur l'URL suivante:

V 9.0.4 IBM MQ 9.0.4 et versions ultérieures :

```
https://localhost:9443/ibmmq/rest/v1/admin/installation
```

IBM MQ 9.0.3 et versions antérieures :

```
https://localhost:9443/ibmmq/rest/v1/installation
```

Vous pouvez utiliser administrative REST API avec l'implémentation HTTP/REST d'un langage de programmation, ou à l'aide d'outils tels que cURL, ou d'un module complémentaire de navigateur client REST.

Pour plus d'informations, voir [administrative REST API](#)

Langage de contrôle IBM i (CL)

IBM i

Cette langue peut être utilisée pour émettre des commandes d'administration à IBM MQ for IBM i. Les commandes peuvent être émises à partir de la ligne de commande ou en écrivant un programme CL. Ces commandes exécutent des fonctions similaires aux commandes PCF, mais le format est différent. Les commandes CL sont conçues exclusivement pour les serveurs et les réponses CL sont lisibles par l'utilisateur, alors que les commandes PCF sont indépendantes de la plateforme et que les formats de commande et de réponse sont destinés à être utilisés par les programmes.

Pour plus de détails sur le langage de contrôle IBM i , voir [Commandes CL IBM MQ for IBM i](#).

IBM MQ Explorer

Windows **Linux**

A l'aide de l' IBM MQ Explorer, vous pouvez effectuer les actions suivantes:

- Définissez et contrôlez diverses ressources, telles que les gestionnaires de files d'attente, les files d'attente, les définitions de processus, les listes de noms, les canaux, les canaux de connexion client, les programmes d'écoute, les services et les clusters.
- Démarrez ou arrêtez un gestionnaire de files d'attente local et ses processus associés.
- Affichez les gestionnaires de files d'attente et les objets associés sur votre poste de travail ou à partir d'autres postes de travail.
- Vérifiez le statut des gestionnaires de files d'attente, des clusters et des canaux.
- Vérifiez les applications, les utilisateurs ou les canaux pour lesquels une file d'attente particulière est ouverte, à partir du statut de la file d'attente.

Sur les systèmes Windows et Linux , vous pouvez démarrer IBM MQ Explorer à l'aide du menu système, du fichier exécutable MQExplorer ou de la commande **strmqcfcg** .

Linux

Sous Linux, pour démarrer correctement IBM MQ Explorer, vous devez pouvoir écrire un fichier dans votre répertoire de base et ce dernier doit exister.

Pour plus d'informations, voir [«Administration à l'aide de IBM MQ Explorer»](#), à la page 135.

Vous pouvez utiliser IBM MQ Explorer pour administrer les gestionnaires de files d'attente éloignées sur d'autres plateformes, y compris z/OS, pour plus de détails et pour télécharger SupportPac MS0T, voir <https://www.ibm.com/support/docview.wss?uid=swg24021041>.

IBM MQ Console

V 9.0.1

Vous pouvez utiliser IBM MQ Console pour administrer IBM MQ à partir d'un navigateur Web.

Pour plus d'informations, voir [«Administration à l'aide de IBM MQ Console»](#), à la page 95.

Application de configuration par défaut de Windows

Windows

Vous pouvez utiliser le programme de configuration par défaut Windows pour créer un ensemble d'objets IBM MQ *de démarrage* (ou par défaut). Un récapitulatif des objets par défaut créés est répertorié dans le [Tableau 1: Objets créés par l' Windows application de configuration par défaut](#).

Microsoft Cluster Service (MSCS)

Windows

Microsoft Cluster Service (MSCS) vous permet de connecter des serveurs à un *cluster*, ce qui offre une meilleure disponibilité des données et des applications et facilite la gestion du système. MSCS détecte automatiquement les échecs de serveurs ou d'applications et initie alors les restaurations requises.

Il est important de ne pas confondre les clusters au sens MSCS avec les clusters IBM MQ. La distinction est la suivante:

Clusters IBM MQ

Il s'agit de groupes de deux ou plusieurs gestionnaires de files d'attente sur un ou plusieurs ordinateurs, fournissant une interconnexion automatique et permettant le partage de files d'attente entre eux pour l'équilibrage de charge et la redondance.

Clusters MSCS

Il s'agit de groupes d'ordinateurs qui sont connectés entre eux et configurés de manière à ce que, en cas de défaillance, MSCS effectue une *reprise en ligne*, en transférant les données d'état des applications de l'ordinateur défaillant vers un autre ordinateur du cluster et en relayant leur fonctionnement.

La [prise en charge de Microsoft Cluster Service \(MSCS\)](#) fournit des informations détaillées sur la configuration de votre système IBM MQ for Windows pour l'utilisation de MSCS.

Information associée

[Comparaison des ensembles de commandes](#)

[Présentation technique du IBM MQ](#)

[Planification](#)

[Configuration](#)

Administration locale et à distance

Vous pouvez administrer les objets IBM MQ en local ou à distance.

administration locale

L'administration locale consiste à exécuter des tâches d'administration sur tous les gestionnaires de files d'attente que vous avez définis sur votre système local. Vous pouvez accéder à d'autres systèmes, par exemple via le programme d'émulation de terminal TCP/IP **telnet**, et y effectuer l'administration. Dans IBM MQ, vous pouvez considérer cela comme une administration locale car aucun canal n'est impliqué, c'est-à-dire que la communication est gérée par le système d'exploitation.

Administration à distance

IBM MQ prend en charge l'administration à partir d'un point de contact unique via l'administration à distance. L'administration à distance vous permet d'émettre des commandes à partir de votre système local qui sont traitées sur un autre système et s'applique également à IBM MQ Explorer. Par exemple, vous pouvez émettre une commande distante pour modifier une définition de file d'attente sur un gestionnaire de files d'attente éloignées. Vous n'avez pas besoin de vous connecter à ce système, bien que les canaux appropriés doivent être définis. Le gestionnaire de files d'attente et le serveur de commandes sur le système cible doivent être en cours d'exécution.

Certaines commandes ne peuvent pas être émises de cette manière, notamment la création ou le démarrage de gestionnaires de files d'attente et de serveurs de commandes. Pour effectuer ce type de tâche, vous devez vous connecter au système distant et émettre les commandes à partir de celui-ci ou créer un processus qui peut émettre les commandes pour vous. Cette restriction s'applique également à IBM MQ Explorer.

«Administration des objets IBM MQ éloignés», à la page 199 décrit plus en détail le sujet de l'administration à distance.

Administration d' IBM MQ à l'aide de commandes de contrôle

Les commandes de contrôle permettent d'effectuer un certain nombre de tâches d'administration IBM MQ sur UNIX, Linux, and Windows.

Si vous souhaitez exécuter des commandes de contrôle, votre ID utilisateur doit être membre du groupe mqm pour la plupart des commandes de contrôle. Pour plus d'informations à ce sujet, voir [Droit d'administration d' IBM MQ sur UNIX, Linux, and Windows](#). En outre, notez les informations spécifiques à l'environnement pour la ou les plateformes utilisées par votre entreprise.

Lorsque vous utilisez des commandes de contrôle qui fonctionnent sur un gestionnaire de files d'attente, vous devez utiliser la commande de l'installation associée au gestionnaire de files d'attente que vous utilisez.

Lors de l'utilisation de commandes de contrôle qui fonctionnent sur un gestionnaire de files d'attente configuré pour utiliser l'authentification de connexion avec CHCKLOCL (REQUIRED), et un échec de connexion est observé:

- Indiquez un ID utilisateur et un mot de passe si la commande de contrôle le permet.
- Utilisez les équivalents MQSC des commandes de contrôle lorsqu'elles existent.
- Démarrez le gestionnaire de files d'attente à l'aide de l'option -ns, alors que les commandes de contrôle qui ne peuvent pas se connecter doivent être exécutées.

Pour la liste complète des commandes de contrôle, voir [Les commandes de contrôle IBM MQ](#).

Utilisation des commandes de contrôle sur les systèmes Windows

Windows

Dans IBM MQ for Windows, vous entrez des commandes de contrôle dans une invite de commande.

Les commandes de contrôle et leurs indicateurs ne sont pas sensibles à la casse, mais les arguments de ces commandes, tels que les noms de file d'attente et les noms de gestionnaire de files d'attente, sont sensibles à la casse.

Par exemple, dans la commande:

```
crtmqm /u SYSTEM.DEAD.LETTER.QUEUE jupiter.queue.manager
```

- Le nom de la commande peut être entré en majuscules ou en minuscules, ou un mélange des deux. Ces éléments sont tous valides: `crtmqm`, `CRTMQMet` `CRTmqm`.
- L'indicateur peut être entré sous la forme `-u`, `-U`, `/uou` `/U`.
- `SYSTEM.DEAD.LETTER.QUEUE` et `jupiter.queue.manager` doivent être entrés exactement comme indiqué.

Utilisation des commandes de contrôle sur les systèmes UNIX et Linux



Sur les systèmes IBM MQ for UNIX et Linux, vous entrez des commandes de contrôle dans une fenêtre shell.

Dans les environnements UNIX, les commandes de contrôle, y compris le nom de la commande elle-même, les indicateurs et tous les arguments, sont sensibles à la casse. Par exemple, dans la commande:

```
crtmqm -u SYSTEM.DEAD.LETTER.QUEUE jupiter.queue.manager
```

- Le nom de la commande doit être `crtmqmet` non `CRTMQM`.
- L'indicateur doit être `-u`, et non `-U`.
- La file d'attente de rebut est appelée `SYSTEM.DEAD.LETTER.QUEUE`.
- L'argument est spécifié sous la forme `jupiter.queue.manager`, qui est différente de `JUPITER.queue.manager`.

Prenez soin de saisir les commandes exactement telles que vous les voyez dans les exemples.

Information associée

[Informations de référence sur les commandes de contrôle IBM MQ](#)

Administration de MQ à l'aide des commandes MQSC

Comment utiliser les commandes MQSC pour effectuer des tâches courantes.

Les commandes MQSC sont disponibles sur toutes les plateformes, y compris IBM i et z/OS.

Vous pouvez utiliser les commandes MQSC pour gérer les objets de gestionnaire de files d'attente, notamment le gestionnaire de files d'attente lui-même, les files d'attente, les définitions de processus, les canaux, les canaux de connexion client, les programmes d'écoute, les services, les listes de noms, les clusters et les objets d'informations d'authentification. Cette section traite des gestionnaires de files d'attente, des files d'attente et des définitions de processus. Pour une présentation des objets canal, canal de connexion client et programme d'écoute, voir [Objets](#). Pour plus d'informations sur toutes les commandes MQSC de gestion des objets de gestionnaire de files d'attente, voir [«Commandes de script \(MQSC\)»](#), à la page 11.

Vous émettez des commandes MQSC vers un gestionnaire de files d'attente à l'aide de la commande **runmqsc**. (Pour plus de détails sur cette commande, voir [runmqsc](#).) Vous pouvez effectuer cette opération de manière interactive, en émettant des commandes à partir d'un clavier, ou vous pouvez rediriger l'unité d'entrée standard (`stdin`) pour exécuter une séquence de commandes à partir d'un fichier texte ASCII. Dans les deux cas, le format des commandes est identique. (Pour plus d'informations sur l'exécution des commandes à partir d'un fichier texte, voir [«Exécution de commandes MQSC à partir de fichiers texte»](#), à la page 16.)

Vous pouvez exécuter la commande **runmqsc** de trois manières, en fonction des indicateurs définis dans la commande:

- Vérifiez une commande sans l'exécuter, où les commandes MQSC sont vérifiées sur un gestionnaire de files d'attente local, mais ne sont pas exécutées.
- Exécutez une commande sur un gestionnaire de files d'attente local, où les commandes MQSC sont exécutées sur un gestionnaire de files d'attente local.
- Exécutez une commande sur un gestionnaire de files d'attente éloignées, où les commandes MQSC sont exécutées sur un gestionnaire de files d'attente éloignées.

Vous pouvez également exécuter la commande suivie d'un point d'interrogation pour afficher la syntaxe.

Les attributs d'objet spécifiés dans les commandes MQSC sont affichés dans cette section en majuscules (par exemple, RQMNAME), bien qu'ils ne soient pas sensibles à la casse. Les noms d'attribut de commande MQSC sont limités à huit caractères.

V 9.0.1 Pour Continuous Delivery, à partir de IBM MQ 9.0.1, vous pouvez définir une invite de votre choix à l'aide de la variable d'environnement **MQPROMPT**. En plus du texte en clair, la variable **MQPROMPT** permet également d'insérer des variables d'environnement, à l'aide de +VARNAME+ notation, de la même manière que les définitions d'objet de service IBM MQ (voir «Définition d'un objet de service», à la page 186). Exemple :

```
sh> export MQPROMPT="+USER+ @ +QMNAME+ @ +MQ_HOST_NAME+> "  
sh> runmqsc MY.QMGR  
5724-H72 (C) Copyright IBM Corp. 1994, 2023.  
Starting MQSC for queue manager MY.QMGR.  
username @ MY.QMGR @ aix1> DISPLAY QMSTATUS
```

V 9.0.0.1 Depuis la IBM MQ 9.0.0 Fix Pack 1, la variable d'environnement **MQPROMPT** est également disponible dans l'édition Long Term Support.

Les commandes MQSC sont détaillées dans la section [MQSC commands](#).

Windows **Linux** Sur les plateformes Windows ou Linux (x86 et x86-64), vous pouvez également effectuer les opérations décrites dans cette section à l'aide de la IBM MQ Explorer. Pour plus d'informations, voir «Administration à l'aide de IBM MQ Explorer», à la page 135.

Information associée

[runmqsc \(exécution des commandes MQSC\)](#)

[Référence MQSC](#)

Commandes de script (MQSC)

Les commandes MQSC fournissent une méthode uniforme d'émission de commandes lisibles par l'utilisateur sur les plateformes IBM MQ.

Le format général des commandes est décrit dans [Les commandes MQSC](#).

Vous devez respecter les règles suivantes lorsque vous utilisez des commandes MQSC:

- Chaque commande commence par un paramètre primaire (un verbe), suivi d'un paramètre secondaire (un nom). Il est ensuite suivi du nom ou du nom générique de l'objet (entre parenthèses) s'il en existe un, qui figure dans la plupart des commandes. Ensuite, les paramètres peuvent généralement se produire dans n'importe quel ordre ; si un paramètre a une valeur correspondante, la valeur doit se produire directement après le paramètre auquel il se rapporte.

Remarque : **z/OS** Sous z/OS, le paramètre secondaire n'a pas besoin d'être second.

- Les mots clés, les parenthèses et les valeurs peuvent être séparés par un nombre quelconque de blancs et de virgules. Une virgule affichée dans les diagrammes de syntaxe peut toujours être remplacée par un ou plusieurs blancs. Au moins un blanc doit précéder immédiatement chaque paramètre (après le paramètre principal) sauf sur z/OS.


- N'importe quel nombre de blancs peut apparaître au début ou à la fin de la commande et entre les paramètres, la ponctuation et les valeurs. Par exemple, la commande suivante est valide:

```
ALTER QLOCAL ('Account' ) TRIGDPTH ( 1)
```


Les blancs à l'intérieur d'une paire de guillemets sont significatifs.

- Des virgules supplémentaires peuvent apparaître partout où des blancs sont autorisés et sont traitées comme s'il s'agissait de blancs (à moins, bien sûr, qu'elles se trouvent dans des chaînes entre guillemets).
- Les paramètres répétés ne sont pas autorisés. La répétition d'un paramètre avec sa version "NO", comme dans REPLACE NOREPLACE, n'est pas non plus autorisée.
- Chaînes contenant des blancs, des caractères minuscules ou des caractères spéciaux autres que:
 - le point (.)
 - la barre oblique (/)
 - le caractère de soulignement (_)
 - le symbole de pourcentage (%)


doivent être placés entre guillemets simples, sauf s'ils sont:

-  Délivré à partir des panneaux d'opérations et de contrôle IBM MQ for z/OS
- Les valeurs génériques se terminant par un astérisque (sous IBM i , elles doivent être placées entre guillemets simples)
- Un astérisque unique, par exemple, TRACE (*) (sous IBM i , ils doivent être placés entre guillemets simples)
- Une spécification de plage contenant un signe deux-points, par exemple, CLASS (01:03)

Si la chaîne elle-même contient une apostrophe, celle-ci est représentée par deux apostrophes. Les caractères minuscules qui ne figurent pas entre guillemets sont convertis en majuscules.

-  Sous Multiplateformes, une chaîne ne contenant aucun caractère (c'est-à-dire deux guillemets simples sans espace entre eux) est interprétée comme un espace vide entre guillemets simples, c'est-à-dire interprété de la même manière que ("). La seule exception à cette règle est si l'attribut utilisé est l'un des suivants:
 - TOPICSTR
 - SUB
 - USERDATA
 - SELECTOR

Deux guillemets simples sans espace sont interprétés comme une chaîne de longueur nulle.

 Sous z/OS, si vous souhaitez placer un espace entre apostrophes, vous devez l'entrer en tant que tel (" "). Une chaîne ne contenant aucun caractère ("") est identique à la chaîne entrée ().

- Dans IBM WebSphere MQ 7.0, les blancs de fin des attributs de chaîne basés sur les types MQCHARV, tels que SELECTOR, les données de sous-utilisateur, sont traités comme significatifs, ce qui signifie que 'abc' n'est pas égal à 'abc'.
- Une parenthèse gauche suivie d'une parenthèse droite, sans aucune information significative entre les deux, par exemple

```
NAME ( )
```

n'est pas valide, sauf indication contraire.

- Les mots clés ne sont pas sensibles à la casse: ALTER, alter et ALTER sont tous acceptables. Tout ce qui n'est pas entre guillemets est plié en majuscules.

- Des synonymes sont définis pour certains paramètres. Par exemple, DEF est toujours un synonyme de DEFINE, de sorte que DEF QLOCAL est valide. Cependant, les synonymes ne sont pas uniquement des chaînes minimales ; DEFI n'est pas un synonyme valide de DEFINE.

Remarque : Il n'existe aucun synonyme pour le paramètre DELETE. Cela permet d'éviter la suppression accidentelle d'objets lors de l'utilisation de DEF, synonyme de DEFINE.

Pour une présentation de l'utilisation des commandes MQSC pour l'administration de IBM MQ, voir [«Administration de MQ à l'aide des commandes MQSC»](#), à la page 10.

Les commandes MQSC utilisent certains caractères spéciaux pour avoir certaines significations. Pour plus d'informations sur ces caractères spéciaux et sur leur utilisation, voir [Caractères ayant une signification particulière](#).

Pour savoir comment générer des scripts à l'aide de commandes MQSC, voir [Génération de scripts de commande](#).

Pour la liste complète des commandes MQSC, voir [Les commandes MQSC](#).

Vous pouvez émettre cette commande depuis les sources 2CR. Pour une explication des symboles de source, voir [Utilisation des commandes sous z/OS](#).

Concepts associés

[«Introduction à IBM MQ Programmable Command Formats»](#), à la page 22

Les formats PCF (Programmable Command Formats) définissent les messages de commande et de réponse qui peuvent être échangés entre un programme et n'importe quel gestionnaire de files d'attente (qui prend en charge les PCF) d'un réseau. Les fichiers PCF simplifient l'administration du gestionnaire de files d'attente et d'autres fonctions d'administration du réseau. Ils peuvent être utilisés pour résoudre le problème de l'administration complexe des réseaux distribués, en particulier à mesure que les réseaux augmentent en taille et en complexité.

Information associée

[Génération de scripts de commande](#)

Noms d'objet IBM MQ dans les commandes MQSC

Comment utiliser les noms d'objet dans les commandes MQSC.

Dans les exemples, nous utilisons des noms longs pour les objets. Il s'agit de vous aider à identifier le type d'objet que vous traitez.

Lorsque vous émettez des commandes MQSC, vous devez spécifier uniquement le nom local de la file d'attente. Dans nos exemples, nous utilisons des noms de file d'attente tels que:

```
ORANGE . LOCAL . QUEUE
```

La partie LOCAL . QUEUE du nom illustre le fait que cette file d'attente est une file d'attente locale. Il n'est **pas** requis pour les noms des files d'attente locales en général.

Nous utilisons également le nom saturn . queue . manager comme nom de gestionnaire de files d'attente. La partie queue . manager du nom illustre le fait que cet objet est un gestionnaire de files d'attente. Il n'est pas requis pour les noms des gestionnaires de files d'attente en général.

Respect de la casse dans les commandes MQSC

Les commandes MQSC, y compris leurs attributs, peuvent être écrites en majuscules ou en minuscules. Les noms d'objet dans les commandes MQSC sont convertis en majuscules (c'est-à-dire que les files d'attente et les files d'attente ne sont pas différenciées), sauf si les noms sont placés entre apostrophes. Si des guillemets ne sont pas utilisés, l'objet est traité avec un nom en majuscules. Pour plus d'informations, voir [Caractères avec des significations spéciales](#).

L'appel de la commande `runmqsc`, comme toutes les commandes de contrôle IBM MQ, est sensible à la casse dans certains environnements IBM MQ. Pour plus d'informations, voir «Administration d' IBM MQ à l'aide de commandes de contrôle», à la page 9.

Entrée et sortie standard

L' *unité d'entrée standard*, également appelée `stdin`, est l'unité à partir de laquelle l'entrée dans le système est effectuée. Il s'agit généralement du clavier, mais vous pouvez spécifier que l'entrée doit provenir d'un port série ou d'un fichier disque, par exemple. L' *unité de sortie standard*, également appelée `stdout`, est l'unité à laquelle la sortie du système est envoyée. Il s'agit généralement d'un affichage, mais vous pouvez rediriger la sortie vers un port série ou un fichier.

Sur les commandes de système d'exploitation et les commandes de contrôle IBM MQ, l'opérateur `<` redirige l'entrée. Si cet opérateur est suivi d'un nom de fichier, l'entrée est extraite du fichier. De même, l'opérateur `>` redirige la sortie ; si cet opérateur est suivi d'un nom de fichier, la sortie est dirigée vers ce fichier.

Utilisation des commandes MQSC en mode interactif

Vous pouvez utiliser les commandes MQSC de manière interactive à l'aide d'une fenêtre de commande ou d'un interpréteur de commandes.

Pour utiliser les commandes MQSC de manière interactive, ouvrez une fenêtre de commande ou un shell et entrez :

```
runmqsc
```

Dans cette commande, aucun nom de gestionnaire de files d'attente n'a été spécifié, de sorte que les commandes MQSC sont traitées par le gestionnaire de files d'attente par défaut. Si vous souhaitez utiliser un gestionnaire de files d'attente différent, indiquez le nom du gestionnaire de files d'attente dans la commande **runmqsc**. Par exemple, pour exécuter des commandes MQSC sur le gestionnaire de files d'attente `jupiter.queue.manager`, utilisez la commande suivante :

```
runmqsc jupiter.queue.manager
```

Après cela, toutes les commandes MQSC que vous entrez sont traitées par ce gestionnaire de files d'attente, en supposant qu'il se trouve sur le même noeud et qu'il est déjà en cours d'exécution.

Vous pouvez maintenant entrer n'importe quelle commande MQSC, selon les besoins. Par exemple, essayez celui-ci :

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE)
```

Pour les commandes comportant trop de paramètres pour tenir sur une seule ligne, utilisez des caractères de continuation pour indiquer qu'une commande se poursuit sur la ligne suivante :

- Le signe moins (-) indique que la commande doit être poursuivie à partir du début de la ligne suivante.
- Un signe plus (+) indique que la commande doit être poursuivie à partir du premier caractère non blanc sur la ligne suivante.

L'entrée de commande se termine par le caractère final d'une ligne non vide qui n'est pas un caractère de continuation. Vous pouvez également arrêter l'entrée de commande de manière explicite en entrant un point-virgule (;). (Cela est particulièrement utile si vous entrez accidentellement un caractère de continuation à la fin de la dernière ligne de l'entrée de commande.)

Commentaires en retour des commandes MQSC

Lorsque vous émettez des commandes MQSC, le gestionnaire de files d'attente renvoie des messages d'opérateur qui confirment vos actions ou vous indiquent les erreurs que vous avez commises. Exemple :

```
AMQ8006: IBM MQ queue created.
```

Ce message confirme qu'une file d'attente a été créée.

```
AMQ8405: Syntax error detected at or near end of command segment below:-
```

```
AMQ8426: Valid MQSC commands are:
```

```
ALTER  
CLEAR  
DEFINE  
DELETE  
DISPLAY  
END  
PING  
REFRESH  
RESET  
RESOLVE  
RESUME  
START  
STOP  
SUSPEND  
4 : end
```

Ce message indique que vous avez fait une erreur de syntaxe.

Ces messages sont envoyés au périphérique de sortie standard. Si vous n'avez pas entré la commande correctement, reportez-vous à la section [Commandes MQSC](#) pour connaître la syntaxe correcte.

Arrêt de l'entrée interactive des commandes MQSC

Pour arrêter l'utilisation des commandes MQSC, entrez la commande END.

Vous pouvez également utiliser le caractère EOF pour votre système d'exploitation.

Concepts associés

«Exécution de commandes MQSC à partir de fichiers texte», à la page 16

L'exécution interactive de commandes MQSC convient aux tests rapides, mais si vous disposez de commandes très longues ou que vous utilisez une séquence de commandes particulière à plusieurs reprises, envisagez de rediriger `stdin` à partir d'un fichier texte.

Information associée

[runmqsc](#)

La commande `runmqsc` sous UNIX et Linux

La ligne de commande `runmqsc` sous UNIX et Linux prend en charge les touches de commande de rappel de commande, d'exécution de commande et Emacs.

Les fonctions suivantes de l'éditeur de ligne de commande sont disponibles:

- Rappel des commandes précédemment entrées à l'aide des touches de déplacement vers le haut et vers le bas
- Exécution automatique du mot clé suivant d'une commande à l'aide de la touche de tabulation et de la barre d'espace
- Touches de commande Emacs ou fonctions de touche de commande similaires

Pour utiliser ces fonctions, la bibliothèque `curses` doit être installée. Si la bibliothèque `curses` n'est pas installée sur votre système, `runmqsc` ne dispose pas des fonctions de l'éditeur de ligne de commande et un message s'affiche lorsque la ligne de commande `runmqsc` est démarrée. Le nom de la bibliothèque `curses` à installer dépend de la plateforme UNIX :

-  Sous AIX, installez `curses`

- **HP-UX** Sous HP-UX, installez Xcurses
- Sur toutes les autres plateformes UNIX et sur Linux, installez ncurses

Personnalisation des liaisons de touches Emacs

Vous pouvez personnaliser les clés qui sont liées aux commandes. Par exemple, vous pouvez lier les touches à des liaisons vi au lieu des liaisons de touches Emacs par défaut.

Les clés sont personnalisées en éditant le fichier `.editrc` qui est stocké dans le répertoire de base. Pour plus d'informations, voir [editrc](#) dans les pages d'aide FreeBSD .

Désactivation du rappel de commande, de l'exécution de commande et des touches de commande Emacs

Vous pouvez désactiver le rappel de commande, l'exécution de commande et les touches de commande Emacs en définissant une variable d'environnement. Définissez la variable d'environnement **MQ_OVERRIDE_LIBEDIT_LOAD** sur TRUE.

Cette variable d'environnement peut être utilisée comme solution palliative lorsque **runmqsc** affiche le message d'information suivant:

```
AMQ8521I: Command completion and history unavailable
```

Exécution de commandes MQSC à partir de fichiers texte

L'exécution interactive de commandes MQSC convient aux tests rapides, mais si vous disposez de commandes très longues ou que vous utilisez une séquence de commandes particulière à plusieurs reprises, envisagez de rediriger `stdin` à partir d'un fichier texte.

Pour rediriger `stdin` à partir d'un fichier texte, créez d'abord un fichier texte contenant les commandes MQSC à l'aide de votre éditeur de texte habituel, puis exécutez la commande **runmqsc** .

Remarque : Si vous exécutez la commande **runmqsc** en mode client en redirigeant `stdin` à partir d'un fichier texte, IBM MQ s'attend à ce que la première ligne du fichier d'entrée soit un mot de passe.

Lorsque vous utilisez la commande **runmqsc** , utilisez les opérateurs de redirection. Par exemple, la commande suivante exécute une séquence de commandes contenues dans le fichier texte `myprog.in`:

```
runmqsc < myprog.in
```

De même, vous pouvez également rediriger la sortie vers un fichier. Un fichier contenant les commandes MQSC pour l'entrée est appelé fichier de commandes MQSC. Le fichier de sortie contenant les réponses du gestionnaire de files d'attente est appelé fichier de sortie.

Pour rediriger à la fois `stdin` et `stdout` dans la commande **runmqsc** , utilisez la forme suivante de la commande:

```
runmqsc < myprog.in > myprog.out
```

Cette commande appelle les commandes MQSC contenues dans le fichier de commandes MQSC `myprog.in` . Etant donné que nous n'avons pas spécifié de nom de gestionnaire de files d'attente, les commandes MQSC s'exécutent sur le gestionnaire de files d'attente par défaut. La sortie est envoyée au fichier texte `myprog.out` . La [Figure 1](#), à la [page 17](#) montre un extrait du fichier de commandes MQSC `myprog.in` et la [Figure 2](#), à la [page 18](#) montre l'extrait correspondant de la sortie dans `myprog.out` .

Pour rediriger `stdin` et `stdout` dans la commande `runmqsc`, pour un gestionnaire de files d'attente (`saturn.queue.manager`) qui n'est pas la valeur par défaut, utilisez la commande suivante:

```
runmqsc saturn.queue.manager < myprog.in > myprog.out
```

Fichiers de commandes MQSC

Les commandes MQSC sont écrites dans un format lisible par l'utilisateur, c'est-à-dire en texte ASCII. Figure 1, à la page 17 est un extrait d'un fichier de commandes MQSC affichant une commande MQSC (**DEFINE QLOCAL**) avec ses attributs. [Commandes MQSC](#) contient une description de chaque commande MQSC et sa syntaxe.

```
.  
. .  
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) REPLACE +  
DESCR(' ') +  
PUT(ENABLED) +  
DEFPRTY(0) +  
DEFPSIST(NO) +  
GET(ENABLED) +  
MAXDEPTH(5000) +  
MAXMSGL(1024) +  
DEFSOPT(SHARED) +  
NOHARDENBO +  
USAGE(NORMAL) +  
NOTRIGGER;  
. .
```

Figure 1. Extraction à partir d'un fichier de commandes MQSC

Pour la portabilité entre les environnements IBM MQ, limitez la longueur de ligne dans les fichiers de commandes MQSC à 72 caractères. Le signe plus indique que la commande se poursuit sur la ligne suivante.

Rapports sur les commandes MQSC

La commande `runmqsc` renvoie un rapport, qui est envoyé à `stdout`. Le rapport contient:

- En-tête identifiant les commandes MQSC comme source du rapport:

```
Starting MQSC for queue manager jupiter.queue.manager.
```

Où `jupiter.queue.manager` est le nom du gestionnaire de files d'attente.

- Liste numérotée facultative des commandes MQSC émises. Par défaut, le texte de l'entrée est répercuté dans la sortie. Dans cette sortie, chaque commande est préfixée par un numéro de séquence, comme illustré dans la [Figure 2](#), à la page 18. Toutefois, vous pouvez utiliser l'indicateur `-e` dans la commande `runmqsc` pour supprimer la sortie.
- Message d'erreur de syntaxe pour toutes les commandes détectées comme étant erronées.
- Message opérateur indiquant le résultat de l'exécution de chaque commande. Par exemple, le message de l'opérateur indiquant que l'exécution d'une commande **DEFINE QLOCAL** a abouti est le suivant:

```
AMQ8006: IBM MQ queue created.
```

- Autres messages résultant d'erreurs générales lors de l'exécution du fichier script.

- Bref récapitulatif statistique du rapport indiquant le nombre de commandes lues, le nombre de commandes comportant des erreurs de syntaxe et le nombre de commandes qui n'ont pas pu être traitées.

Remarque : Le gestionnaire de files d'attente tente de traiter uniquement les commandes qui ne comportent aucune erreur de syntaxe.

```
Starting MQSC for queue manager jupiter.queue.manager.
.
.
12:  DEFINE QLOCAL('ORANGE.LOCAL.QUEUE') REPLACE +
:    DESCR(' ') +
:    PUT(ENABLED) +
:    DEFPRTY(0) +
:    DEFPSIST(NO) +
:    GET(ENABLED) +
:    MAXDEPTH(5000) +
:    MAXMSGL(1024) +
:    DEFSOPT(SHARED) +
:    NOHARDENBO +
:    USAGE(NORMAL) +
:    NOTRIGGER;
AMQ8006: IBM MQ queue created.
.
.
.
```

Figure 2. Extraction à partir d'un fichier de rapport de commandes MQSC

Exécution des fichiers de commandes MQSC fournis

Les fichiers de commandes MQSC suivants sont fournis avec IBM MQ:

amqscos0.tst

Définitions des objets utilisés par les exemples de programme.

amqscic0.tst

Définitions des files d'attente pour les transactions CICS .

Windows Dans IBM MQ for Windows, ces fichiers se trouvent dans le répertoire `MQ_INSTALLATION_PATH\tools\mqsc\samples`. `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.

Linux **UNIX** Sur les systèmes UNIX and Linux , ces fichiers se trouvent dans le répertoire `MQ_INSTALLATION_PATH/samp`. `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.

La commande qui les exécute est la suivante:

```
runmqsc < amqscos0.tst >test.out
```

Utilisation de runmqsc pour vérifier les commandes

Vous pouvez utiliser la commande **runmqsc** pour vérifier les commandes MQSC sur un gestionnaire de files d'attente local sans les exécuter. Pour ce faire, définissez l'indicateur **-v** dans la commande **runmqsc** , par exemple:

```
runmqsc -v < myprog.in > myprog.out
```

Lorsque vous appelez **runmqsc** dans un fichier de commandes MQSC, le gestionnaire de files d'attente vérifie chaque commande et renvoie un rapport sans exécuter réellement les commandes MQSC.

Cela vous permet de vérifier la syntaxe des commandes dans votre fichier de commandes. Ceci est particulièrement important si vous êtes:

- Exécution d'un grand nombre de commandes à partir d'un fichier de commandes.
- Utilisation d'un fichier de commandes MQSC plusieurs fois.

Le rapport renvoyé est similaire à celui présenté dans la [Figure 2](#), à la [page 18](#).

Vous ne pouvez pas utiliser cette méthode pour vérifier les commandes MQSC à distance. Par exemple, si vous tentez d'exécuter cette commande:

```
runmqsc -w 30 -v jupiter.queue.manager < myprog.in > myprog.out
```

L'indicateur **-w**, que vous utilisez pour indiquer que le gestionnaire de files d'attente est distant, est ignoré et la commande est exécutée localement en mode de vérification. 30 est le nombre de secondes pendant lequel IBM MQ attend les réponses du gestionnaire de files d'attente éloignées.

Concepts associés

«[Entrée et sortie standard](#)», à la [page 14](#)

L'*unité d'entrée standard*, également appelée `stdin`, est l'unité à partir de laquelle l'entrée dans le système est effectuée. Il s'agit généralement du clavier, mais vous pouvez spécifier que l'entrée doit provenir d'un port série ou d'un fichier disque, par exemple. L'*unité de sortie standard*, également appelée `stdout`, est l'unité à laquelle la sortie du système est envoyée. Il s'agit généralement d'un affichage, mais vous pouvez rediriger la sortie vers un port série ou un fichier.

«[Utilisation des commandes MQSC en mode interactif](#)», à la [page 14](#)

Vous pouvez utiliser les commandes MQSC de manière interactive à l'aide d'une fenêtre de commande ou d'un interpréteur de commandes.

Information associée

[runmqsc](#)

Exécution de commandes MQSC à partir de fichiers de traitement par lots

Si vous avez des commandes très longues ou que vous utilisez une séquence de commandes particulière à plusieurs reprises, envisagez de rediriger `stdin` à partir d'un fichier de traitement par lots.

Pour rediriger `stdin` à partir d'un fichier de traitement par lots, créez d'abord un fichier de traitement par lots contenant les commandes MQSC à l'aide de votre éditeur de texte habituel. Lorsque vous utilisez la commande `runmqsc`, utilisez les opérateurs de redirection. L'exemple suivant :

1. Crée un gestionnaire de files d'attente de test, TESTQM
2. Crée un CLNTCONN et un programme d'écoute correspondant pour utiliser le port TCP/IP 1600
3. Crée une file d'attente de test, TESTQ
4. Insère un message dans la file d'attente à l'aide de l'exemple de programme `amqspuic`

```

export MYTEMPQM=TESTQM
export MYPOR=1600
export MQCHLLIB=/var/mqm/qmgrs/$MQTEMPQM/@ipcc

crtmqm $MYTEMPQM
stimqm $MYTEMPQM
runmqtsr -m $MYTEMPQM -t TCP -p $MYPOR &

runmqsc $MYTEMPQM << EOF
DEFINE CHANNEL(NTLM) CHLTYPE(SVRCONN) TRPTYPE(TCP)
DEFINE CHANNEL(NTLM) CHLTYPE(CLNTCONN) QMNAME('$MYTEMPQM') CONNAME('hostname($MYPOR)')
ALTER CHANNEL(NTLM) CHLTYPE(CLNTCONN)
DEFINE QLOCAL(TESTQ)
EOF

amqsputc TESTQ $MYTEMPQM << EOF
hello world
EOF

endmqm -i $MYTEMPQM

```

Figure 3. Exemple de script pour l'exécution de commandes MQSC à partir d'un fichier de traitement par lots

Résolution des problèmes liés aux commandes MQSC

Si vous ne parvenez pas à exécuter des commandes MQSC, utilisez les informations de cette rubrique pour voir si l'un de ces problèmes communs s'applique à vous. Il n'est pas toujours évident de savoir quel est le problème lorsque vous lisez l'erreur générée par une commande.

Lorsque vous utilisez la commande `runmqsc`, tenez compte des points suivants:

- Utilisez l'opérateur `<` pour rediriger l'entrée à partir d'un fichier. Si vous omettez cet opérateur, le gestionnaire de files d'attente interprète le nom de fichier comme un nom de gestionnaire de files d'attente et émet le message d'erreur suivant:

```
AMQ8118E: IBM MQ queue manager does not exist.
```

- Si vous redirigez la sortie vers un fichier, utilisez l'opérateur de redirection `>`. Par défaut, le fichier est placé dans le répertoire de travail en cours lors de l'appel de `runmqsc`. Indiquez un nom de fichier qualifié complet pour envoyer votre sortie vers un fichier et un répertoire spécifiques.
- Vérifiez que vous avez créé le gestionnaire de files d'attente qui va exécuter les commandes, en utilisant la commande suivante pour afficher tous les gestionnaires de files d'attente:

```
dspmq
```

- Il doit être en cours d'exécution. Si tel n'est pas le cas, démarrez-le (voir [Démarrage d'un gestionnaire de files d'attente](#)). Un message d'erreur s'affiche si vous tentez de démarrer un gestionnaire de files d'attente déjà en cours d'exécution.
- Indiquez un nom de gestionnaire de files d'attente dans la commande `runmqsc` si vous n'avez pas défini de gestionnaire de files d'attente par défaut ou si vous obtenez l'erreur suivante:

```
AMQ8146E: IBM MQ queue manager not available.
```

- Vous ne pouvez pas spécifier une commande MQSC en tant que paramètre de la commande `runmqsc`. Par exemple, ceci n'est pas valide:

```
runmqsc DEFINE QLOCAL(FRED)
```

- Vous ne pouvez pas entrer de commandes MQSC avant d'exécuter la commande `runmqsc`.

- Vous ne pouvez pas exécuter de commandes de contrôle à partir de **runmqsc**. Par exemple, vous ne pouvez pas exécuter la commande `strmqm` pour démarrer un gestionnaire de files d'attente alors que vous exécutez des commandes MQSC de manière interactive. Dans ce cas, vous recevez des messages d'erreur similaires aux suivants:

```
runmqsc
.
.
Starting MQSC for queue manager jupiter.queue.manager.

1 : strmqm saturn.queue.manager
AMQ8405: Syntax error detected at or near end of cmd segment below:-s

AMQ8426: Valid MQSC commands are:
ALTER
CLEAR
DEFINE
DELETE
DISPLAY
END
PING
REFRESH
RESET
RESOLVE
RESUME
START
STOP
SUSPEND
2 : end
```

Automatisation de l'administration de IBM MQ à l'aide des commandes PCF

Vous pouvez décider qu'il serait avantageux pour votre installation d'automatiser certaines tâches d'administration et de surveillance. Vous pouvez automatiser les tâches d'administration pour les gestionnaires de files d'attente locales et éloignées à l'aide des commandes PCF (Programmable Command Format). Cette section suppose que vous avez l'expérience de l'administration des objets IBM MQ .

Commandes PCF

Les commandes PCF (IBM MQ programmable command format) peuvent être utilisées pour programmer des tâches d'administration dans un programme d'administration. Ainsi, à partir d'un programme, vous pouvez manipuler des objets de gestionnaire de files d'attente (files d'attente, définitions de processus, listes de noms, canaux, canaux de connexion client, programmes d'écoute, services et objets d'informations d'authentification) et même manipuler les gestionnaires de files d'attente eux-mêmes.

Les commandes PCF couvrent la même plage de fonctions que les commandes MQSC. Vous pouvez écrire un programme pour émettre des commandes PCF vers n'importe quel gestionnaire de files d'attente du réseau à partir d'un noeud unique. Ainsi, vous pouvez à la fois centraliser et automatiser les tâches d'administration.

Chaque commande PCF est une structure de données imbriquée dans la partie données d'application d'un message IBM MQ . Chaque commande est envoyée au gestionnaire de files d'attente cible à l'aide de la fonction MQI MQPUT de la même manière que tout autre message. A condition que le serveur de commandes soit en cours d'exécution sur le gestionnaire de files d'attente qui reçoit le message, le serveur de commandes l'interprète comme un message de commande et exécute la commande. Pour obtenir les réponses, l'application émet un appel MQGET et les données de réponse sont renvoyées dans une autre structure de données. L'application peut ensuite traiter la réponse et agir en conséquence.

Remarque : Contrairement aux commandes MQSC, les commandes PCF et leurs réponses ne sont pas au format texte que vous pouvez lire.

En bref, voici quelques-uns des éléments nécessaires à la création d'un message de commande PCF:

Descripteur de message

Il s'agit d'un descripteur de message IBM MQ standard, dans lequel:

- Le type de message (*MsgType*) est MQMT_REQUEST.
- Le format de message (*Format*) est MQFMT_ADMIN.

Données d'application

Contient le message PCF incluant l'en-tête PCF, dans lequel:

- Le type de message PCF (*Type*) indique MQCFT_COMMAND.
- L'identificateur de la commande indique la commande, par exemple, *Change Queue* (MQCMD_CHANGE_Q).

Pour obtenir une description complète des structures de données PCF et savoir comment les implémenter, voir [«Introduction à IBM MQ Programmable Command Formats»](#), à la page 22.

Attributs d'objet PCF

Les attributs d'objet dans PCF ne sont pas limités à huit caractères, comme pour les commandes MQSC. Ils sont présentés dans ce guide en italique. Par exemple, l'équivalent PCF de RQMNAME est *RemoteQMGrName*.



Echapper les fichiers PCF

Les fichiers PCF d'arrêt programme sont des commandes PCF qui contiennent des commandes MQSC dans le texte du message. Vous pouvez utiliser des fichiers PCF pour envoyer des commandes à un gestionnaire de files d'attente éloignées. Pour plus d'informations sur les fichiers PCF d'échappement, voir [Echap](#).

Introduction à IBM MQ Programmable Command Formats

Les formats PCF (Programmable Command Formats) définissent les messages de commande et de réponse qui peuvent être échangés entre un programme et n'importe quel gestionnaire de files d'attente (qui prend en charge les PCF) d'un réseau. Les fichiers PCF simplifient l'administration du gestionnaire de files d'attente et d'autres fonctions d'administration du réseau. Ils peuvent être utilisés pour résoudre le problème de l'administration complexe des réseaux distribués, en particulier à mesure que les réseaux augmentent en taille et en complexité.

Les formats de commande programmables sont pris en charge par:

-  IBM MQ for AIX
-  IBM MQ for HP-UX
-  IBM MQ for IBM i
-  IBM MQ for Linux
-  IBM MQ for Solaris
-  IBM MQ for Windows
-  IBM MQ for z/OS

Résolution des incidents liés aux commandes PCF

L'administration des réseaux distribués peut devenir complexe. Les problèmes d'administration continuent de croître à mesure que les réseaux augmentent en taille et en complexité.

Exemples d'administration spécifique à la messagerie et à la mise en file d'attente:

- Gestion des ressources.

Par exemple, la création et la suppression de files d'attente.

- Surveillance des performances.

Par exemple, la longueur maximale de la file d'attente ou le débit des messages.

- Contrôle.

Par exemple, l'optimisation des paramètres de file d'attente tels que la longueur maximale de la file d'attente, la longueur maximale des messages et l'activation et la désactivation des files d'attente.

- Routage des messages.

Définition de routes alternatives à travers un réseau.

Les commandes PCF IBM MQ peuvent être utilisées pour simplifier l'administration du gestionnaire de files d'attente et d'autres fonctions d'administration du réseau. Les commandes PCF vous permettent d'utiliser une application unique pour effectuer l'administration du réseau à partir d'un seul gestionnaire de files d'attente au sein du réseau.

Que sont les PCF?

Les PCF définissent les messages de commande et de réponse qui peuvent être échangés entre un programme et n'importe quel gestionnaire de files d'attente (qui prend en charge les PCF) d'un réseau. Vous pouvez utiliser des commandes PCF dans un programme d'application de gestion des systèmes pour l'administration des objets IBM MQ : objets d'informations d'authentification, canaux, programmes d'écoute de canal, listes de noms, définitions de processus, gestionnaires de files d'attente, files d'attente, services et classes de stockage. L'application peut fonctionner à partir d'un point unique du réseau pour communiquer des informations de commande et de réponse avec n'importe quel gestionnaire de files d'attente, local ou distant, à l'aide du gestionnaire de files d'attente local.


Chaque gestionnaire de files d'attente possède une file d'attente d'administration avec un nom de file d'attente standard et votre application peut envoyer des messages de commande PCF à cette file d'attente. Chaque gestionnaire de files d'attente dispose également d'un serveur de commandes pour traiter les messages de commande de la file d'attente d'administration. Les messages de commande PCF peuvent donc être traités par n'importe quel gestionnaire de files d'attente du réseau et les données de réponse peuvent être renvoyées à votre application à l'aide de la file d'attente de réponses spécifiée. Les commandes PCF et les messages de réponse sont envoyés et reçus à l'aide de l'interface MQI (Message Queue Interface) normale.

Pour obtenir la liste des commandes PCF disponibles, y compris leurs paramètres, voir [Définitions des formats de commande programmables](#).

Utilisation des formats de commande programmables IBM MQ

Vous pouvez utiliser des fichiers PCF dans un programme de gestion des systèmes pour l'administration à distance d' IBM MQ .

Cette section contient :

- [«Messages de commande PCF»](#), à la page 23
- [«Réponses PCF dans IBM MQ»](#), à la page 26
-  [«Réponses étendues»](#), à la page 28
- [Règles d'appellation des objets IBM MQ](#)
- [«Vérification des droits d'accès pour les commandes PCF dans IBM MQ»](#), à la page 30

Messages de commande PCF

Les messages de commande PCF sont constitués d'un en-tête PCF, de paramètres identifiés dans cet en-tête et de données de message définies par l'utilisateur. Les messages sont émis à l'aide d'appels d'interface de file d'attente de messages.

Chaque commande et ses paramètres sont envoyés sous la forme d'un message de commande distinct contenant un en-tête PCF suivi d'un certain nombre de structures de paramètres ; pour plus de détails sur

l'en-tête PCF, voir [MQCFH-PCF header](#) et pour un exemple de structure de paramètre, voir [MQCFST-PCF string parameter](#). L'en-tête PCF identifie la commande et le nombre de structures de paramètres qui suivent dans le même message. Chaque structure de paramètres fournit un paramètre à la commande.

Les réponses aux commandes, générées par le serveur de commandes, ont une structure similaire. Il existe un en-tête PCF, suivi d'un certain nombre de structures de paramètres. Les réponses peuvent consister en plusieurs messages, mais les commandes consistent toujours en un seul message.

Multi Sous [Multiplateformes](#), la file d'attente à laquelle les commandes PCF sont envoyées est toujours appelée `SYSTEM.ADMIN.COMMAND.QUEUE`.

z/OS Sous z/OS, les commandes sont envoyées à `SYSTEM.COMMAND.INPUT`, bien que `SYSTEM.ADMIN.COMMAND.QUEUE` peut être un alias pour lui. Le serveur de commandes gérant cette file d'attente envoie les réponses à la file d'attente définie par les zones *ReplyToQ* et *ReplyToQMGr* dans le descripteur de message du message de commande.

Comment émettre des messages de commande PCF

Utilisez les appels MQI (Message Queue Interface) normaux, `MQPUT`, `MQGET`, etc., pour insérer et extraire des messages de commande et de réponse PCF vers et depuis leurs files d'attente.

Remarque :

Vérifiez que le serveur de commandes est en cours d'exécution sur le gestionnaire de files d'attente cible pour que la commande PCF puisse être traitée sur ce gestionnaire de files d'attente.

Pour obtenir la liste des fichiers d'en-tête fournis, voir [IBM MQ COPY, header, include et module files](#).

Descripteur de message pour une commande PCF

Le descripteur de message IBM MQ est intégralement documenté dans [MQMD-Descripteur de message](#).

Un message de commande PCF contient les zones suivantes dans le descripteur de message:

Rapport

Toute valeur valide, selon les besoins.

MsgType

Cette zone doit être `MQMT_REQUEST` pour indiquer un message nécessitant une réponse.

Expiration

Toute valeur valide, selon les besoins.

Commentaires

Défini sur `MQFB_NONE`

Codage

Si vous envoyez des données à l'un des systèmes suivants, définissez cette zone sur le codage utilisé pour les données de message ; la conversion est effectuée si nécessaire:

- **IBM i** IBM i
- **Linux** Linux
- **UNIX** UNIX
- **Windows** Windows

CodedCharSetId

Si vous effectuez un envoi vers l'un des systèmes suivants, définissez cette zone sur l'identificateur de jeu de caractères codés utilisé pour les données du message ; la conversion est effectuée si nécessaire:

- **IBM i** IBM i

-  Linux
-  UNIX
-  Windows

Format

Défini sur MQFMT_ADMIN.

Priorité

Toute valeur valide, selon les besoins.

Persistance

Toute valeur valide, selon les besoins.

MsgId

L'application émettrice peut spécifier n'importe quelle valeur ou MQMI_NONE peut être spécifié pour demander au gestionnaire de files d'attente de générer un identificateur de message unique.

CorrelId

L'application émettrice peut spécifier n'importe quelle valeur ou MQCI_NONE peut être spécifié pour indiquer qu'il n'y a pas d'identificateur de corrélation.

ReplyToQ

Nom de la file d'attente de réception de la réponse.

ReplyToQMGr

Nom du gestionnaire de files d'attente pour la réponse (ou vide).

Zones de contexte de message

Ces zones peuvent être définies sur n'importe quelle valeur valide, selon les besoins. Normalement, l'option d'insertion de message MQPMO_DEFAULT_CONTEXT est utilisée pour définir les zones de contexte de message sur les valeurs par défaut.

Si vous utilisez une structure MQMD version-2 , vous devez définir les zones supplémentaires suivantes:

GroupId

Défini sur MQGI_NONE

MsgSeqNumber

Définir sur 1

Décalage

Définir sur 0

MsgFlags

Défini sur MQMF_NONE

OriginalLength

Défini sur MQOL_UNDEFINED

Envoi de données utilisateur

Les structures PCF peuvent également être utilisées pour envoyer des données de message définies par l'utilisateur. Dans ce cas, la zone *Format* du descripteur de message doit être définie sur MQFMT_PCF.

Envoi et réception de messages PCF dans une file d'attente spécifiée

Envoi de messages PCF à une file d'attente spécifiée

Pour envoyer un message à une file d'attente spécifiée, l'appel de sac mqPut convertit le contenu du sac spécifié en message PCF et envoie le message à la file d'attente spécifiée. Le contenu du sac est inchangé après l'appel.

Comme entrée pour cet appel, vous devez fournir:

- Un descripteur de connexion MQI.
- Descripteur d'objet de la file d'attente dans laquelle le message doit être placé.
- Descripteur de message. Pour plus d'informations sur le descripteur de message, voir [MQMD-Descripteur de message](#).
- Options d'insertion de message à l'aide de la structure MQPMO. Pour plus d'informations sur la structure MQPMO, voir [MQPMO-Options d'insertion de message](#).
- Poignée du sac à convertir en message.

Remarque : Si le sac contient un message d'administration et que l'appel d'interrogation mqAdda été utilisé pour insérer des valeurs dans le sac, la valeur de l'élément de données MQIASY_COMMAND doit être une commande INQUIRE reconnue par MQAI.

Pour une description complète de l'appel de sac mqPut, voir [mqPutBag](#).

Réception de messages PCF d'une file d'attente spécifiée

Pour recevoir un message d'une file d'attente spécifiée, l'appel mqGetBag extrait un message PCF d'une file d'attente spécifiée et convertit les données de message en un sac de données.

Comme entrée pour cet appel, vous devez fournir:

- Un descripteur de connexion MQI.
- Descripteur d'objet de la file d'attente à partir de laquelle le message doit être lu.
- Descripteur de message. Dans la structure MQMD, le paramètre **Format** doit être MQFMT_ADMIN, MQFMT_EVENT ou MQFMT_PCF.

Remarque : Si le message est reçu dans une unité de travail (c'est-à-dire avec l'option MQGMO_SYNCPOINT) et que le format du message n'est pas pris en charge, l'unité de travail peut être annulée. Le message est ensuite réintégré dans la file d'attente et peut être extrait à l'aide de l'appel MQGET au lieu de l'appel Bag mqGet. Pour plus d'informations sur le descripteur de message, voir [Options MQGMO-Get-message](#).

- Obtenir les options de message à l'aide de la structure MQGMO. Pour plus d'informations sur la structure MQGMO, voir [MQMD-Descripteur de message](#).
- Poignée du sac devant contenir le message converti.

Pour une description complète de l'appel de sac mqGet, voir [mqGetBag](#).

Réponses PCF dans IBM MQ

En réponse à chaque commande, le serveur de commandes génère un ou plusieurs messages de réponse. Un message de réponse a un format similaire à celui d'un message de commande.

L'en-tête PCF possède la même valeur d'identificateur de commande que la commande à laquelle il correspond à une réponse (voir [MQCFH-en-tête PCF](#) pour plus de détails). L'identificateur de message et l'identificateur de corrélation sont définis en fonction des options de rapport de la demande.

Si le type d'en-tête PCF du message de commande est MQCFT_COMMAND, seules les réponses standard sont générées. Ces commandes sont prises en charge sur toutes les plateformes, à l'exception de z/OS. Les anciennes applications ne prennent pas en charge PCF sous z/OS ; IBM MQ Explorer sur Windows est l'une de ces applications (toutefois, la IBM WebSphere MQ 6.0 ou version ultérieure IBM MQ Explorer prend en charge PCF sur z/OS).

Si le type d'en-tête PCF du message de commande est MQCFT_COMMAND_XR, des réponses étendues ou standard sont générées. Ces commandes sont prises en charge sur z/OS et sur d'autres plateformes. Les commandes émises sur z/OS génèrent uniquement des réponses étendues. Sur d'autres plateformes, l'un ou l'autre type de réponse peut être généré.

Si une seule commande spécifie un nom d'objet générique, une réponse distincte est renvoyée dans son propre message pour chaque objet correspondant. Pour la génération de réponse, une seule commande

avec un nom générique est traitée comme plusieurs commandes individuelles (à l'exception de la zone de contrôle MQCFC_LAST ou MQCFC_NOT_LAST). Sinon, un message de commande génère un message de réponse.

Certaines réponses PCF peuvent renvoyer une structure même si elle n'est pas demandée. Cette structure est illustrée dans la définition de la réponse (Définitions des formats de commande programmables) comme *toujours renvoyé*. La raison pour laquelle, pour ces réponses, il est nécessaire de nommer les objets dans la réponse pour identifier l'objet auquel les données s'appliquent.

Descripteur de message pour une réponse

Un message de réponse comporte les zones suivantes dans le descripteur de message:

MsgType

Cette zone est MQMT_REPLY.

MsgId

Cette zone est générée par le gestionnaire de files d'attente.

CorrelId

Cette zone est générée en fonction des options de rapport du message de commande.

Format

Cette zone est MQFMT_ADMIN.

Codage

Défini sur MQENC_NATIVE.

CodedCharSetId

Défini sur MQCCSI_Q_MGR.

Persistance

Identique à celui du message de commande.

Priorité

Identique à celui du message de commande.

La réponse est générée avec MQPMO_PASS_IDENTITY_CONTEXT.

Réponses standard

Les messages de commande avec un type d'en-tête MQCFT_COMMAND, des réponses standard sont générées. Ces commandes sont prises en charge sur toutes les plateformes, à l'exception de z/OS.

Il existe trois types de réponse standard:

- Réponse OK
- Réponse d'erreur
- Réponse aux données

Réponse OK

Cette réponse consiste en un message commençant par un en-tête de format de commande, avec une zone *CompCode* de type MQCC_OK ou MQCC_WARNING.

Pour MQCC_OK, *Reason* est MQRC_NONE.

Pour MQCC_WARNING, *Reason* identifie la nature de l'avertissement. Dans ce cas, l'en-tête de format de commande peut être suivi d'une ou de plusieurs structures de paramètres d'avertissement appropriées à ce code anomalie.

Dans les deux cas, pour une commande d'interrogation, d'autres structures de paramètres peuvent suivre, comme décrit dans les sections suivantes.

Réponse d'erreur

Si la commande comporte une erreur, un ou plusieurs messages de réponse d'erreur sont envoyés (plusieurs peuvent être envoyés même pour une commande qui n'aurait normalement qu'un seul message de réponse). MQCFC_LAST ou MQCFC_NOT_LAST est défini comme approprié pour ces messages de réponse d'erreur.

Chaque message de ce type commence par un en-tête de format de réponse, avec une valeur *CompCode* de MQCC_FAILED et une zone *Reason* qui identifie l'erreur particulière. En général, chaque message décrit une erreur différente. En outre, chaque message a soit zéro, soit une (jamais plus d'une) structure de paramètre d'erreur après l'en-tête. Cette structure de paramètre, s'il en existe une, est une structure MQCFIN, avec une zone *Parameter* contenant l'un des éléments suivants:

- ID_PARAMÈTRE_MQIACF_ID

La zone *Value* de la structure correspond à l'identificateur du paramètre erroné (par exemple, MQCA_Q_NAME).

- ID_ERREUR_MQIACF

Cette valeur est utilisée avec une valeur *Reason* (dans l'en-tête de format de commande) de MQRC_UNEXPECTED_ERROR. La zone *Value* de la structure MQCFIN correspond au code anomalie inattendu reçu par le serveur de commandes.

- MQIACF_SELECTOR

Cette valeur se produit si une structure de liste (MQCFIL) envoyée avec la commande contient un sélecteur en double ou non valide. La zone *Reason* dans l'en-tête de format de commande identifie l'erreur et la zone *Value* dans la structure MQCFIN est la valeur de paramètre dans la structure MQCFIL de la commande qui était erronée.

- MQIACF_DÉCALAGE_ERREUR_ERREUR

Cette valeur se produit en cas d'erreur de comparaison de données dans la commande Ping Channel. La zone *Value* de la structure correspond au décalage de l'erreur de comparaison du canal ping.

- MQIA_CODED_CHAR_SET_ID

Cette valeur se produit lorsque l'identificateur de jeu de caractères codés dans le descripteur de message du message de commande PCF entrant ne correspond pas à celui du gestionnaire de files d'attente cible. La zone *Value* de la structure correspond à l'identificateur de jeu de caractères codés du gestionnaire de files d'attente.

Le dernier (ou seul) message de réponse d'erreur est une réponse récapitulative, avec une zone *CompCode* de MQCC_FAILED et une zone *Reason* de MQRCCF_COMMAND_FAILED. Ce message n'a pas de structure de paramètre à la suite de l'en-tête.

Réponse aux données

Cette réponse consiste en une réponse OK (comme décrit précédemment) à une commande d'interrogation. La réponse OK est suivie de structures supplémentaires contenant les données demandées, comme décrit dans la rubrique Définitions des formats de commande programmables.

Les applications ne doivent pas dépendre du renvoi de ces structures de paramètres supplémentaires dans un ordre particulier.

Réponses étendues

Les commandes émises sur z/OS génèrent des réponses étendues.

Il existe trois types de réponse étendue:

- Réponse du message, avec le type MQCFT_XR_MSG
- Réponse d'élément, avec le type MQCFT_XR_ITEM
- Réponse récapitulative, avec le type MQCFT_XR_SUMMARY

Chaque commande peut générer un ou plusieurs ensembles de réponses. Chaque ensemble de réponses comprend un ou plusieurs messages, numérotés séquentiellement à partir de 1 dans le champ *MsgSeqNumber* de l'en-tête PCF. La zone *Control* de la dernière (ou seule) réponse de chaque ensemble a la valeur MQCFC_LAST. Pour toutes les autres réponses de l'ensemble, cette valeur est MQCFC_NOT_LAST.

Toute réponse peut inclure une ou plusieurs structures MQCFBS facultatives dans lesquelles la zone *Parameter* est définie sur MQBACF_RESPONSE_SET, la valeur étant un identificateur d'ensemble de réponses. Les identificateurs sont uniques et identifient l'ensemble des réponses qui contiennent la réponse. Pour chaque ensemble de réponses, une structure MQCFBS l'identifie.

Les réponses étendues ont au moins deux structures de paramètres:

- Une structure MQCFBS avec la zone *Parameter* définie sur MQBACF_RESPONSE_ID. La valeur de cette zone est l'identificateur de l'ensemble de réponses auquel appartient la réponse. L'identificateur du premier ensemble est arbitraire. Dans les ensembles suivants, l'identificateur est précédemment notifié dans une structure MQBACF_RESPONSE_SET.
- Une structure MQCFST avec la zone *Parameter* définie sur MQCACF_RESPONSE_Q_MGR_NAME, la valeur étant le nom du gestionnaire de files d'attente d'où provient l'ensemble de réponses.

De nombreuses réponses ont des structures de paramètres supplémentaires, et ces structures sont décrites dans les sections suivantes.

Vous ne pouvez pas déterminer à l'avance le nombre de réponses dans un ensemble autre qu'en obtenant des réponses jusqu'à ce qu'une réponse avec MQCFC_LAST soit trouvée. Vous ne pouvez pas non plus déterminer à l'avance combien d'ensembles de réponses il y a, car n'importe quel ensemble peut inclure des structures MQBACF_RESPONSE_SET pour indiquer que des ensembles supplémentaires sont générés.

Réponses étendues aux commandes Inquire

Les commandes d'interrogation génèrent normalement une réponse d'élément (type MQCFT_XR_ITEM) pour chaque élément trouvé qui correspond aux critères de recherche spécifiés. L'en-tête de la réponse d'élément comporte une zone *CompCode* avec la valeur MQCC_OK et une zone *Reason* avec la valeur MQRC_NONE. Il inclut également d'autres structures de paramètres décrivant l'élément et ses attributs demandés, comme décrit dans [Définitions des formats de commande programmables](#).

Si un élément est en erreur, la zone *CompCode* de l'en-tête a la valeur MQCC_FAILED et la zone *Reason* identifie l'erreur particulière. Des structures de paramètres supplémentaires sont incluses pour identifier l'élément.

Certaines commandes Inquire peuvent renvoyer des réponses de message générales (non spécifiques à un nom) en plus des réponses d'élément. Ces réponses sont des réponses d'information ou d'erreur de type MQCFT_XR_MSG.

Si la commande Inquire aboutit, il peut éventuellement y avoir une réponse récapitulative (type MQCFT_XR_SUMMARY), avec une valeur *CompCode* de MQCC_OK et une valeur de zone *Reason* de MQRC_NONE.

Si la commande Inquire échoue, des réponses d'élément peuvent être renvoyées et il peut éventuellement y avoir une réponse récapitulative (type MQCFT_XR_SUMMARY), avec une valeur *CompCode* de MQCC_FAILED et une valeur de zone *Reason* de MQRCCF_COMMAND_FAILED.

Réponses étendues aux commandes autres que l'interrogation

Les commandes réussies génèrent des réponses de message dans lesquelles la zone *CompCode* de l'en-tête a la valeur MQCC_OK et la zone *Reason* a la valeur MQRC_NONE. Il existe toujours au moins un message ; il peut s'agir d'un message d'information (MQCFT_XR_MSG) ou d'un récapitulatif (MQCFT_XR_SUMMARY). Il peut éventuellement y avoir des messages d'information supplémentaires (type MQCFT_XR_MSG). Chaque message d'information peut inclure un certain nombre de structures de paramètres supplémentaires avec des informations sur la commande ; voir les descriptions de commande individuelles pour les structures qui peuvent se produire.

Les commandes qui échouent génèrent des réponses de message d'erreur (type MQCFT_XR_MSG), dans lesquelles la zone *CompCode* de l'en-tête a la valeur MQCC_FAILED et la zone *Reason* identifie l'erreur particulière. Chaque message peut inclure un certain nombre de structures de paramètres supplémentaires avec des informations sur l'erreur: voir les descriptions d'erreur individuelles pour les structures qui peuvent se produire. Des réponses de message d'information peuvent être générées. Il peut éventuellement y avoir une réponse récapitulative (MQCFT_XR_SUMMARY), avec une valeur *CompCode* de MQCC_FAILED et une valeur de zone *Reason* de MQRCCF_COMMAND_FAILED.

Réponses étendues aux commandes utilisant CommandScope

Si une commande utilise le paramètre **CommandScope** ou génère une commande à l'aide du paramètre **CommandScope**, il existe un ensemble de réponses initial du gestionnaire de files d'attente dans lequel la commande a été reçue. Ensuite, un ou plusieurs ensembles de réponses distincts sont générés pour chaque gestionnaire de files d'attente vers lequel la commande est dirigée (comme si plusieurs commandes individuelles étaient émises). Enfin, il existe un ensemble de réponses du gestionnaire de files d'attente de réception qui inclut une réponse récapitulative globale (type MQCFT_XR_SUMMARY). La structure du paramètre MQCACF_RESPONSE_Q_MGR_NAME identifie le gestionnaire de files d'attente qui génère chaque ensemble.


L'ensemble de réponses initial comporte les structures de paramètres supplémentaires suivantes:

- MQIACF_COMMAND_INFO (MQCFIN). Les valeurs possibles dans cette structure sont MQCMDI_CMDScope_ACCEPTED ou MQCMDI_CMDScope_GENERATED.
- MQIACF_CMDScope_Q_MGR_COUNT (MQCFIN). Cette structure indique le nombre de gestionnaires de files d'attente auxquels la commande est envoyée.

Vérification des droits d'accès pour les commandes PCF dans IBM MQ

Lorsqu'une commande PCF est traitée, le *UserIdentifier* du descripteur de message dans le message de commande est utilisé pour les vérifications des droits d'accès aux objets IBM MQ requises. La vérification des droits d'accès est implémentée différemment sur chaque plateforme, comme décrit dans cette rubrique.

Les vérifications sont effectuées sur le système sur lequel la commande est en cours de traitement ; par conséquent, cet ID utilisateur doit exister sur le système cible et disposer des droits requis pour traiter la commande. Si le message provient d'un système distant, l'un des moyens d'obtenir l'ID existant sur le système cible consiste à disposer d'un ID utilisateur correspondant sur les systèmes local et distant.

Remarque :  Pour plus d'informations sur le contrôle des droits d'accès sur z/OS, voir [Tâche 1: Identifier les paramètres système z/OS](#).

IBM MQ for IBM i



Pour pouvoir traiter une commande PCF, l'ID utilisateur doit disposer des droits *dsp* sur l'objet IBM MQ sur le système cible.

En outre, des vérifications des droits d'accès aux objets IBM MQ sont effectuées pour certaines commandes PCF, comme illustré dans la [Tableau 1](#), à la page 31.

Dans la plupart des cas, ces vérifications sont les mêmes que celles effectuées par les commandes CL IBM MQ équivalentes émises sur un système local. Pour plus d'informations sur le mappage des droits IBM MQ vers les droits système IBM i et sur les droits requis pour les commandes CL IBM MQ, voir [Configuration de la sécurité sur IBM i](#). Les détails de la sécurité concernant les exits sont fournis dans la documentation [Link level security using a security exit](#).

Pour traiter l'une des commandes suivantes, l'ID utilisateur doit être membre du profil de groupe QMQMADM:

- Envoyer une commande PING à un canal
- Modifier un canal

- Copier un canal
- Créer un canal
- Supprimer un canal
- Réinitialisation du canal
- Résolution du canal
- Démarrer un canal
- Arrêter le canal
- Démarrer un initialiseur de canal
- Démarrer un programme d'écoute de canaux

IBM MQ for UNIX, Linux, and Windows



Pour traiter une commande PCF, l'ID utilisateur doit disposer des droits *dsp* sur l'objet gestionnaire de files d'attente sur le système cible. En outre, des vérifications des droits d'accès aux objets IBM MQ sont effectuées pour certaines commandes PCF, comme illustré dans la [Tableau 1](#), à la page 31.

Pour traiter l'une des commandes suivantes, l'ID utilisateur doit appartenir au groupe *mqm*.

Remarque : Pour Windows **uniquement**, l'ID utilisateur peut appartenir au groupe *Administrators* ou au groupe *mqm*.

- Modifier un canal
- Copier un canal
- Créer un canal
- Supprimer un canal
- Envoyer une commande PING à un canal
- Réinitialisation du canal
- Démarrer un canal
- Arrêter le canal
- Démarrer un initialiseur de canal
- Démarrer un programme d'écoute de canaux
- Résolution du canal
- Réinitialisation d'un cluster
- Régénérer un cluster
- Interrompre un gestionnaire de files d'attente
- Reprendre un gestionnaire de files d'attente

Droits d'accès aux objets IBM MQ pour Multiplatforms



| Tableau 1. Droits sur les objets | | |
|--|--------------------------------------|---|
| Commande | IBM MQ Les droits d'accès d'un objet | Droits sur les classes (pour le type d'objet) |
| Modifier des informations d'authentification | dsp et chg | Non applicable |
| Modifier un canal | dsp et chg | Non applicable |

Tableau 1. Droits sur les objets (suite)

| Commande | IBM MQ Les droits d'accès d'un objet | Droits sur les classes (pour le type d'objet) |
|---|--------------------------------------|---|
| Modifier le programme d'écoute de canal | dsp et chg | Non applicable |
| Modifier le canal de connexion client | dsp et chg | Non applicable |
| Modifier une liste de noms | dsp et chg | Non applicable |
| Changement de processus | dsp et chg | Non applicable |
| Modifier une file d'attente | dsp et chg | Non applicable |
| Modifier un gestionnaire de files d'attente | chg <i>voir les remarques 3 et 5</i> | Non applicable |
| Modifier le service | dsp et chg | Non applicable |
| Mettre à blanc une file d'attente | clr | Non applicable |
| Copier des informations d'authentification | dsp | crt |
| Copier les informations d'authentification (remplacement) <i>voir la remarque 1</i> | de: dsp à: chg | crt |
| Copier un canal | dsp | crt |
| Copier le canal (remplacer) <i>voir la remarque 1</i> | de: dsp à: chg | crt |
| Programme d'écoute de canal de copie | dsp | crt |
| Programme d'écoute de canal de copie (remplacement) <i>voir la remarque 1</i> | de: dsp à: chg | crt |
| Copier le canal de connexion client | dsp | crt |
| Copier le canal de connexion client (remplacement) <i>voir la remarque 1</i> | de: dsp à: chg | crt |
| Copier une liste de noms | dsp | crt |
| Copier la liste de noms (remplacement) <i>voir la remarque 1</i> | de: dsp à: dsp et chg | crt |
| Copier un processus | dsp | crt |
| Processus de copie (remplacement) <i>voir la remarque 1</i> | de: dsp à: chg | crt |
| Copier une file d'attente | dsp | crt |

Tableau 1. Droits sur les objets (suite)

| Commande | IBM MQ Les droits d'accès d'un objet | Droits sur les classes (pour le type d'objet) |
|--|--|---|
| File d'attente de copie (remplacement) voir la remarque 1 | de: dsp à: dsp et chg | crt |
| Créer des informations d'authentification | (informations d'authentification par défaut du système) dsp | crt |
| Créer des informations d'authentification (remplacer) voir la remarque 1 | (informations d'authentification par défaut du système) dsp sur: chg | crt |
| Créer un canal | (canal par défaut du système) dsp | crt |
| Créer un canal (remplacer) voir la remarque 1 | (canal par défaut du système) dsp vers: chg | crt |
| Créer un programme d'écoute de canal | (programme d'écoute par défaut du système) dsp | crt |
| Créer un programme d'écoute de canal (remplacement) voir la remarque 1 | (programme d'écoute par défaut du système) dsp sur: chg | crt |
| Créer un canal de connexion client | (canal par défaut du système) dsp | crt |
| Créer un canal de connexion client (remplacement) voir la remarque 1 | (canal par défaut du système) dsp vers: chg | crt |
| Créer une liste de noms | (liste de noms par défaut du système) dsp | crt |
| Créer une liste de noms (remplacer) voir la remarque 1 | (liste de noms par défaut du système) dsp sur: dsp et chg | crt |
| Créer un processus | (processus par défaut du système) dsp | crt |
| Créer un processus (remplacer) voir la remarque 1 | (processus par défaut du système) dsp sur: chg | crt |
| Créer une file d'attente | (file d'attente par défaut du système) dsp | crt |
| Créer une file d'attente (remplacement) voir la remarque 1 | (file d'attente par défaut du système) dsp vers: dsp et chg | crt |
| Création de service | (file d'attente par défaut du système) dsp | crt |
| Create Service (Replace) voir la remarque 1 | (file d'attente par défaut du système) dsp vers: chg | crt |
| Supprimer des informations d'authentification | dsp et dlt | Non applicable |
| Supprimer l'enregistrement de droits d'accès | (objet gestionnaire de files d'attente) chg voir la remarque 4 | voir la remarque 4 |
| Supprimer un canal | dsp et dlt | Non applicable |
| Supprimer le programme d'écoute de canal | dsp et dlt | Non applicable |

Tableau 1. Droits sur les objets (suite)

| Commande | IBM MQ Les droits d'accès d'un objet | Droits sur les classes (pour le type d'objet) |
|--|---|---|
| Supprimer le canal de connexion client | dsp et dlt | Non applicable |
| Supprimer une liste de noms | dsp et dlt | Non applicable |
| Supprimer un processus | dsp et dlt | Non applicable |
| Supprimer une file d'attente | dsp et dlt | Non applicable |
| Supprimer le service | dsp et dlt | Non applicable |
| Consulter des informations d'authentification | dsp | Non applicable |
| Consulter des enregistrements de droits | <i>voir la remarque 4</i> | <i>voir la remarque 4</i> |
| Consulter un canal | dsp | Non applicable |
| Consulter le programme d'écoute de canal | dsp | Non applicable |
| Interroger le statut du canal (pour ChannelType MQCHT_CLSSDR) | Inq | Non applicable |
| Consulter le canal de connexion client | dsp | Non applicable |
| Consulter une liste de noms | dsp | Non applicable |
| Consulter un processus | dsp | Non applicable |
| Consulter la file d'attente | dsp | Non applicable |
| Consulter les gestionnaires de files d'attente | <i>voir la remarque 3</i> | Non applicable |
| Consulter le statut d'une file d'attente | dsp | Non applicable |
| Consulter le service | dsp | Non applicable |
| Envoyer une commande PING à un canal | ctrl | Non applicable |
| Envoyer une commande Ping à un gestionnaire de files d'attente | <i>voir la remarque 3</i> | Non applicable |
| Régénérer un gestionnaire de files d'attente | (objet gestionnaire de files d'attente) chg | Non applicable |
| Actualiser la sécurité (pour SecurityType MQSECTYPE_SSL) | (objet gestionnaire de files d'attente) chg | Non applicable |
| Réinitialisation du canal | ctrlx | Non applicable |
| Réinitialiser un gestionnaire de files d'attente | (objet gestionnaire de files d'attente) chg | Non applicable |

Tableau 1. Droits sur les objets (suite)

| Commande | IBM MQ Les droits d'accès d'un objet | Droits sur les classes (pour le type d'objet) |
|--|--|---|
| Réinitialiser les statistiques de file d'attente | dsp et chg | Non applicable |
| Résolution du canal | ctrlx | Non applicable |
| Définir l'enregistrement de droits d'accès | (objet gestionnaire de files d'attente) chg <i>voir la remarque 4</i> | <i>voir la remarque 4</i> |
| Démarrer un canal | ctrl | Non applicable |
| Arrêter le canal | ctrl | Non applicable |
| Arrêter la connexion | (objet gestionnaire de files d'attente) chg | Non applicable |
| Démarrage du programme d'écoute | ctrl | Non applicable |
| Arrêter le programme d'écoute | ctrl | Non applicable |
| Démarrer le service | ctrl | Non applicable |
| Arrêter le service | ctrl | Non applicable |
| Echap | <i>Voir la remarque 2</i> | <i>Voir la remarque 2</i> |

Remarques :

1. Cette commande s'applique si l'objet à remplacer existe, sinon la vérification des droits est celle de la création ou de la copie sans remplacement.
2. Les droits requis sont déterminés par la commande MQSC définie par le texte d'échappement et sont équivalents à l'une des commandes précédentes.
3. Pour traiter une commande PCF, l'ID utilisateur doit disposer des droits dsp sur l'objet gestionnaire de files d'attente sur le système cible.
4. Cette commande PCF est autorisée sauf si le serveur de commandes a été démarré avec le paramètre -a. Par défaut, le serveur de commandes démarre lorsque le gestionnaire de files d'attente est démarré et sans le paramètre -a. Pour plus d'informations, voir [Programmable command formats reference](#).
5. L'octroi à un ID utilisateur de droits d'accès chg pour un gestionnaire de files d'attente permet de définir des enregistrements de droits d'accès pour tous les groupes et utilisateurs. N'accordez pas ce droit à des utilisateurs ou à des applications ordinaires.

IBM MQ fournit également des points d'exit de sécurité de canal afin que vous puissiez fournir vos propres programmes d'exit utilisateur pour le contrôle de la sécurité. Pour plus d'informations, voir [Affichage d'un canal](#).



Utilisation de MQAI pour simplifier l'utilisation des fichiers PCF

L'interface d'administration IBM MQ (MQAI) est une interface de programmation d'IBM MQ disponible sur les systèmes AIX, HP-UX, IBM i, Linux, Solaris, et Windows. Il effectue des tâches d'administration sur un gestionnaire de files d'attente IBM MQ à l'aide de sacs de données pour gérer les propriétés (ou les paramètres) des objets d'une manière plus simple que l'utilisation des formats PCF (Programmable Command Formats).

MQAI effectue des tâches d'administration sur un gestionnaire de files d'attente en utilisant des sacs de données. Les sacs de données vous permettent de gérer les propriétés (ou paramètres) des objets d'une manière plus simple que l'utilisation de fichiers PCF.

Les avantages de l'utilisation de MQAI sont les suivants:

Simplifiez l'utilisation des messages PCF

Le MQAI est un moyen plus simple d'administrer IBM MQ. Si vous utilisez l'interface MQAI, vous n'avez pas besoin d'écrire vos propres messages PCF. Cela évite les problèmes liés aux structures de données complexes.

Pour transmettre des paramètres dans des programmes écrits à l'aide d'appels MQI, le message PCF doit contenir la commande et les détails de la chaîne ou des données entières. Pour créer cette configuration manuellement, vous devez ajouter plusieurs instructions dans votre programme pour chaque structure et vous devez allouer de l'espace mémoire. Cette tâche peut être longue et laborieuse.

Les programmes écrits à l'aide des MQAI passent des paramètres dans le sac de données approprié, et vous n'avez besoin que d'une seule instruction pour chaque structure. L'utilisation des sacs de données MQAI élimine la nécessité pour vous de gérer les grappes et d'allouer du stockage, et fournit un certain degré d'isolement par rapport aux détails du PCF.

Gérer les conditions d'erreur plus facilement

Il est difficile d'obtenir des codes retour à partir des commandes PCF. L'interface MQAI permet au programme de gérer plus facilement les conditions d'erreur.

Echange de données entre des applications

Les données d'application sont envoyées au format PCF et compressées et décompressées par MQAI. Si vos données de message se composent d'entiers et de chaînes de caractères, vous pouvez utiliser l'interface MQAI pour tirer parti de la conversion de données IBM MQ intégrée pour les données PCF. Cela évite d'avoir à écrire des exits de conversion de données.

Une fois que vous avez créé et rempli votre sac de données, vous pouvez envoyer un message de commande d'administration au serveur de commandes d'un gestionnaire de files d'attente à l'aide de l'appel `mqExecute`. Cet appel attend les messages de réponse. L'appel `mqExecute` gère l'échange avec le serveur de commandes et renvoie des réponses dans un *sac de réponse*.

Exemples d'utilisation de MQAI

La liste ci-dessous fournit des exemples de programmes illustrant l'utilisation de MQAI. Les exemples effectuent les tâches suivantes:

1. Création d'une file d'attente locale. [«Exemple de programme C pour la création d'une file d'attente locale \(amqsaicq.c\)»](#), à la page 43
2. Affichez les événements à l'écran à l'aide d'un moniteur d'événements simple. [«Exemple de programme C pour l'affichage d'événements à l'aide d'un moniteur d'événements \(amqsaie.c\)»](#), à la page 46
3. Imprimez la liste de toutes les files d'attente locales et de leur profondeur actuelle. [«Exemple de programme C pour l'interrogation des files d'attente et des informations d'impression \(amqsailq.c\)»](#), à la page 58
4. Imprimez la liste de tous les canaux et de leurs types. [«Exemple de programme C pour l'interrogation d'objets de canal \(amqsaicl.c\)»](#), à la page 53

Génération de votre application MQAI

Pour générer votre application à l'aide de MQAI, vous établissez un lien vers les mêmes bibliothèques que pour IBM MQ. Pour plus d'informations sur la génération de vos applications IBM MQ, voir [Génération d'une application procédurale](#).

Conseils et astuces pour la configuration de IBM MQ à l'aide de MQAI

MQAI utilise des messages PCF pour envoyer des commandes d'administration au serveur de commandes au lieu de traiter directement avec le serveur de commandes lui-même. Des conseils pour la configuration de IBM MQ à l'aide de MQAI sont disponibles dans [«Conseils et astuces pour l'utilisation de MQAI afin de configurer IBM MQ»](#), à la page 37.

Information associée

IBM MQ -Guide de référence de l'interface d'administration

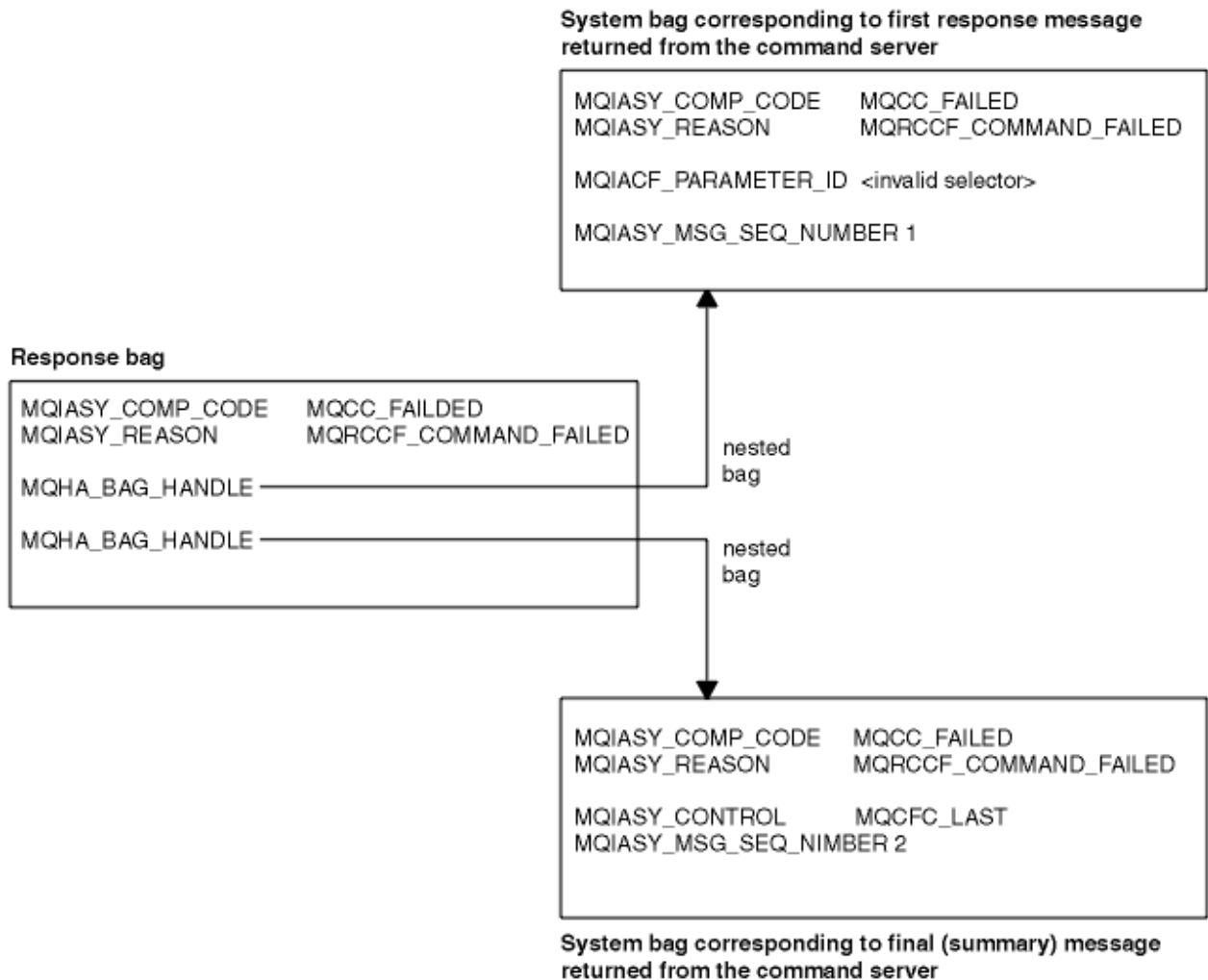
Conseils et astuces pour l'utilisation de MQAI afin de configurer IBM MQ

L'interface d'administration IBM MQ (MQAI) utilise des messages PCF pour envoyer des commandes d'administration au serveur de commandes plutôt que de traiter directement avec le serveur de commandes lui-même. Voici quelques conseils pour configurer IBM MQ à l'aide de MQAI.

- Les chaînes de caractères dans IBM MQ sont remplies de blancs à une longueur fixe. A l'aide de C, les chaînes à terminaison nulle peuvent normalement être fournies en tant que paramètres d'entrée aux interfaces de programmation IBM MQ .
- Pour effacer la valeur d'un attribut de chaîne, définissez-la sur un seul blanc plutôt que sur une chaîne vide.
- Prenez en compte à l'avance les attributs que vous souhaitez modifier et n'examinez que ces attributs.
- Certains attributs ne peuvent pas être modifiés, par exemple un nom de file d'attente ou un type de canal. Veillez à ne modifier que les attributs qui peuvent être modifiés. Reportez-vous à la liste des paramètres obligatoires et facultatifs pour l'objet de modification PCF spécifique. Voir Définitions des formats de commande programmables.
- Si un appel MQAI échoue, certains détails de l'échec sont renvoyés au sac de réponse. Des détails supplémentaires peuvent ensuite être trouvés dans un sac imbriqué accessible par le sélecteur MQHA_BAG_HANDLE. Par exemple, si un appel mqExecute échoue avec le code anomalie MQRCCF_COMMAND_FAILED, ces informations sont renvoyées dans le sac de réponse. Ce code raison peut être dû au fait qu'un sélecteur spécifié n'était pas valide pour le type de message de commande et que ce détail d'informations se trouve dans un sac imbriqué accessible par un descripteur de sac.

Pour plus d'informations sur MQExecute, voir «Envoi de commandes d'administration au serveur de commandes qm à l'aide de l'appel mqExecute», à la page 71

Le diagramme suivant illustre ce scénario:



Rubriques MQAI avancées

Informations sur l'indexation, la conversion de données et l'utilisation du descripteur de message

- Indexation

Les index sont utilisés lors du remplacement ou de la suppression d'éléments de données existants dans un sac afin de préserver l'ordre d'insertion. Des détails complets sur l'indexation sont disponibles dans [«Indexation dans MQAI»](#), à la page 38.

- Conversion de données

Les chaînes contenues dans un sac de données MQAI peuvent se trouver dans divers jeux de caractères codés et peuvent être converties à l'aide de l'appel d'entier mqSet. Pour plus de détails sur la conversion des données, voir [«Traitement de la conversion de données dans MQAI»](#), à la page 39.

- Utilisation du descripteur de message

MQAI génère un descripteur de message qui est défini sur une valeur initiale lors de la création du sac de données. Vous trouverez des détails complets sur l'utilisation du descripteur de message dans [«Utilisation du descripteur de message dans MQAI»](#), à la page 41.

Indexation dans MQAI

Les index sont utilisés lors du remplacement ou de la suppression d'éléments de données existants dans un sac. Il existe trois types d'indexation, qui permettent d'extraire facilement des éléments de données.

Chaque sélecteur et chaque valeur d'un élément de données dans un sac sont associés à trois numéros d'index:

- Index relatif à d'autres éléments ayant le même sélecteur.
- Index relatif à la catégorie du sélecteur (utilisateur ou système) à laquelle appartient l'élément.
- Index relatif à tous les éléments de données du sac (utilisateur et système).

Cela permet l'indexation par des sélecteurs utilisateur, des sélecteurs système, ou les deux, comme illustré dans la [Figure 4](#), à la page 39.

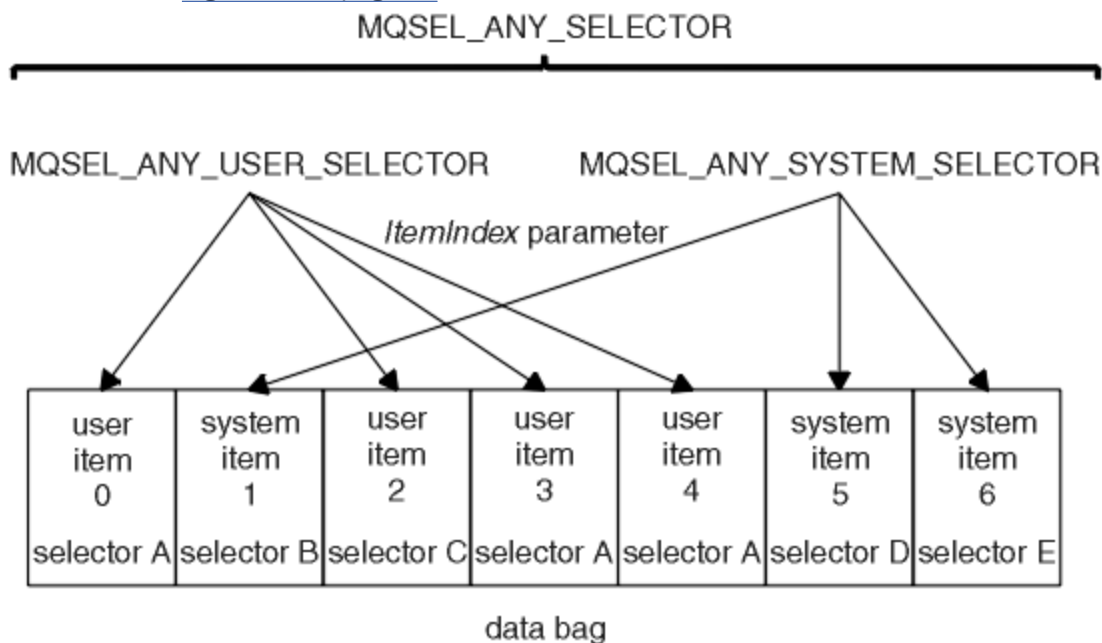


Figure 4. Indexation

Dans la figure [Figure 4](#), à la page 39, l'élément utilisateur 3 (sélecteur A) peut être référencé par les paires d'index suivantes:

| Selector | ItemIndex |
|-------------------------|------------------|
| sélecteur A | 1 |
| MQSEL_ANY_USER_SELECTOR | 2 |
| MQSEL_ANY_SELECTOR | 3 |

L'index est basé sur zéro comme un tableau en C ; s'il y a n'occurrences, l'index est compris entre zéro et 'n-1', sans écart.

Les index sont utilisés lors du remplacement ou de la suppression d'éléments de données existants dans un sac. Lorsqu'il est utilisé de cette manière, l'ordre d'insertion est conservé, mais les index d'autres éléments de données peuvent être affectés. Pour des exemples, voir [Modification des informations dans un sac](#) et [Suppression d'éléments de données](#).

Les trois types d'indexation permettent d'extraire facilement des éléments de données. Par exemple, s'il existe trois instances d'un sélecteur particulier dans un sac, l'appel `mqCountItems` peut compter le nombre d'instances de ce sélecteur, et les appels `mqInquire*` peuvent spécifier à la fois le sélecteur et l'index pour interroger ces valeurs uniquement. Ceci est utile pour les attributs qui peuvent avoir une liste de valeurs telles que certaines des exits sur les canaux.

Traitement de la conversion de données dans MQAI

Les chaînes contenues dans un sac de données MQAI peuvent être dans une variété de jeux de caractères codés. Ces chaînes peuvent être converties à l'aide de l'appel `mqSetInteger`.

Comme les messages PCF, les chaînes contenues dans un sac de données MQAI peuvent être dans une variété de jeux de caractères codés. En général, toutes les chaînes d'un message PCF se trouvent dans le même jeu de caractères codés, c'est-à-dire dans le même jeu que le gestionnaire de files d'attente.

Chaque élément de chaîne d'un sac de données contient deux valeurs: la chaîne elle-même et le CCSID. La chaîne qui est ajoutée au sac est obtenue à partir du paramètre **Buffer** de la chaîne mqAddou de l'appel de chaîne mqSet. Le CCSID est obtenu à partir de l'élément système contenant un sélecteur de MQIASY_CODED_CHAR_SET_ID. Connue sous le nom de *CCSID du sac*, il peut être modifié à l'aide de l'appel mqSetInteger.

Lorsque vous interrogez la valeur d'une chaîne contenue dans un sac de données, le CCSID est un paramètre de sortie de l'appel.

Le Tableau 2, à la page 40 montre les règles appliquées lors de la conversion de sacs de données en messages et vice versa:

| <i>Tableau 2. Traitement CCSID</i> | | | |
|------------------------------------|--|---|---|
| Appel MQAI | CCSID | Entrée à appeler | Sortie à appeler |
| mqBagToBuffer | CCSID du sac (<u>1</u>) | Ignorée | Inchangé |
| mqBagToBuffer | CCSID de chaîne dans le sac | Utilisé | Inchangé |
| mqBagToBuffer | CCSID de chaîne dans la mémoire tampon | Non applicable | Copié à partir des CCSID de chaîne dans le sac |
| mqBufferToBag | CCSID du sac (<u>1</u>) | Ignorée | Inchangé |
| mqBufferToBag | CCSID de chaîne dans la mémoire tampon | Utilisé | Inchangé |
| mqBufferToBag | CCSID de chaîne dans le sac | Non applicable | Copié à partir des CCSID de chaîne dans la mémoire tampon |
| SacmqPut | CCSID MQMD | Utilisé | Inchangé (<u>2</u>) |
| SacmqPut | CCSID du sac (<u>1</u>) | Ignorée | Inchangé |
| SacmqPut | CCSID de chaîne dans le sac | Utilisé | Inchangé |
| SacmqPut | CCSID de chaîne dans le message envoyé | Non applicable | Copié à partir des CCSID de chaîne dans le sac |
| mqGetSac | CCSID MQMD | Utilisé pour la conversion de données de message | Défini sur le CCSID des données renvoyées (<u>3</u>) |
| mqGetSac | CCSID du sac (<u>1</u>) | Ignorée | Inchangé |
| mqGetSac | CCSID de chaîne dans le message | Utilisé | Inchangé |
| mqGetSac | CCSID de chaîne dans le sac | Non applicable | Copié à partir des CCSID de chaîne dans le message |
| mqExecute | CCSID du sac de demande | Utilisé pour le MQMD du message de demande (<u>4</u>) | Inchangé |

| Appel MQAI | CCSID | Entrée à appeler | Sortie à appeler |
|------------|--|---|---|
| mqExecute | CCSID du sac de réponse | Utilisé pour la conversion de données du message de réponse (4) | Défini sur le CCSID des données renvoyées (3) |
| mqExecute | CCSID de chaîne dans le sac de demande | Utilisé pour le message de demande | Inchangé |
| mqExecute | CCSID de chaîne dans le sac de réponse | Non applicable | Copié à partir des CCSID de chaîne dans le message de réponse |

Remarques :

1. Le CCSID du sac est l'élément système avec le sélecteur MQIASY_CODED_CHAR_SET_ID.
2. MQCCSI_Q_MGR est remplacé par le CCSID réel du gestionnaire de files d'attente.
3. Si une conversion de données est demandée, le CCSID des données renvoyées est identique à la valeur de sortie. Si la conversion de données n'est pas demandée, le CCSID des données renvoyées est identique à la valeur du message. Notez qu'aucun message n'est renvoyé si la conversion de données est demandée mais échoue.
4. Si le CCSID est MQCCSI_DEFAULT, le CCSID du gestionnaire de files d'attente est utilisé.

Information associée

[Conversion de données](#)

[Fichier ccsid_part2.tbl](#)

Utilisation du descripteur de message dans MQAI

Le descripteur de message généré par MQAI est défini sur une valeur initiale lors de la création du sac de données.

Le type de commande PCF est obtenu à partir de l'élément système avec le sélecteur MQIASY_TYPE. Lorsque vous créez votre sac de données, la valeur initiale de cet élément est définie en fonction du type de sac que vous créez:

| Type de sac | Valeur initiale de l'élément MQIASY_TYPE |
|-----------------------------------|--|
| MQCBO_ADMIN_BAG | MQCFT_COMMAND (COMMANDE MQ) |
| SAC de commande mqcbo_commande | MQCFT_COMMAND (COMMANDE MQ) |
| MQCBO_* | Utilisateur_MQCF |

Lorsque MQAI génère un descripteur de message, les valeurs utilisées dans les paramètres **Format** et **MsgType** dépendent de la valeur de l'élément système avec le sélecteur MQIASY_TYPE, comme illustré dans la [Tableau 3](#), à la page 41.

| Type de commande PCF | Format | MsgType |
|-----------------------------|-------------|--------------|
| MQCFT_COMMAND (COMMANDE MQ) | MQFMT_ADMIN | MQMT_REQUEST |
| RAPPORT MQCF | MQFMT_ADMIN | MQMT_REPORT |

| Type de commande PCF | Format | MsgType |
|----------------------|-------------|---------------|
| MQCFT_REPONSE | MQFMT_ADMIN | MQMT_REPLY |
| MQCFT_TRACE_ROUTE | MQFMT_ADMIN | MQMT_DATAGRAM |
| MQCFT_EVENT | MQFMT_EVENT | MQMT_DATAGRAM |
| MQCFT_* | MQFMT_PCF | MQMT_DATAGRAM |

Tableau 4, à la page 41 montre que si vous créez un sac d'administration ou un sac de commande, le *Format* du descripteur de message est MQFMT_ADMIN et le *MsgType* est MQMT_REQUEST. Cela convient pour un message de demande PCF envoyé au serveur de commandes lorsqu'une réponse est attendue.

Les autres paramètres du descripteur de message prennent les valeurs indiquées dans le Tableau 5, à la page 42.

| Paramètre | Valeur |
|-----------------------|--|
| <i>StrucId</i> | ID_STRUCD_MQM |
| <i>Version</i> | MQMD_VERSION_1 |
| <i>Report</i> | MQRO_AUCUN |
| <i>MsgType</i> | voir Tableau 4, à la page 41 |
| <i>Expiry</i> | 30 secondes (remarque «1», à la page 42) |
| <i>Feedback</i> | MQFB_AUCUN |
| <i>Encoding</i> | MQENC_NATIVE |
| <i>CodedCharSetId</i> | dépend du CCSID du sac (remarque «2», à la page 42) |
| <i>Format</i> | voir Tableau 4, à la page 41 |
| <i>Priority</i> | MQPRI_PRIORITY_AS_Q_DEF |
| <i>Persistence</i> | MQPER_NON_PERSISTENT |
| <i>MsgId</i> | MQMI_AUCUN |
| <i>CorrelId</i> | MQCI_NONE |
| <i>BackoutCount</i> | 0 |
| <i>ReplyToQ</i> | voir la remarque «3», à la page 42 |
| <i>ReplyToQMgr</i> | blanc |

Remarques :

1. Cette valeur peut être remplacée dans l'appel mqExecute à l'aide du paramètre **OptionsBag** . Pour plus d'informations, voir [mqExecute](#).
2. Voir «Traitement de la conversion de données dans MQAI», à la page 39.
3. Nom de la file d'attente de réponses spécifiée par l'utilisateur ou de la file d'attente dynamique temporaire générée par MQAI pour les messages de type MQMT_REQUEST. Vide dans le cas contraire.

Exemple de programme C pour la création d'une file d'attente locale (amqsaicq.c)

L'exemple de programme C amqsaicq.c crée une file d'attente locale à l'aide de MQAI.

```

/*****
/*
/* Program name: AMQSAICQ.C
/*
/* Description: Sample C program to create a local queue using the
/* IBM MQ Administration Interface (MQAI).
/*
/* Statement: Licensed Materials - Property of IBM
/*
/* 84H2000, 5765-B73
/* 84H2001, 5639-B42
/* 84H2002, 5765-B74
/* 84H2003, 5765-B75
/* 84H2004, 5639-B43
/*
/* (C) Copyright IBM Corp. 1999, 2023.
/*
*****/
/*
/* Function:
/* AMQSAICQ is a sample C program that creates a local queue and is an
/* example of the use of the mqExecute call.
/*
/* - The name of the queue to be created is a parameter to the program.
/*
/* - A PCF command is built by placing items into an MQAI bag.
/* These are:-
/* - The name of the queue
/* - The type of queue required, which, in this case, is local.
/*
/* - The mqExecute call is executed with the command MQCMD_CREATE_Q.
/* The call generates the correct PCF structure.
/* The call receives the reply from the command server and formats into
/* the response bag.
/*
/* - The completion code from the mqExecute call is checked and if there
/* is a failure from the command server then the code returned by the
/* command server is retrieved from the system bag that is
/* embedded in the response bag to the mqExecute call.
/*
/* Note: The command server must be running.
/*
*****/
/*
/* AMQSAICQ has 2 parameters - the name of the local queue to be created
/* - the queue manager name (optional)
/*
*****/
/* Includes
*****/
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

#include <cmqc.h> /* MQI */
#include <cmqcfh.h> /* PCF */
#include <cmqbc.h> /* MQAI */

void CheckCallResult(MQCHAR *, MQLONG , MQLONG );
void CreateLocalQueue(MQHCONN, MQCHAR *);

int main(int argc, char *argv[])
{
    MQHCONN hConn; /* handle to IBM MQ connection */
    MQCHAR QMName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name */
    MQLONG connReason; /* MQCONN reason code */
    MQLONG compCode; /* completion code */
    MQLONG reason; /* reason code */

```

```

/*****
/* First check the required parameters */
/*****
printf("Sample Program to Create a Local Queue\n");
if (argc < 2)
{
    printf("Required parameter missing - local queue name\n");
    exit(99);
}

/*****
/* Connect to the queue manager */
/*****
if (argc > 2)
    strncpy(QMName, argv[2], (size_t)MQ_Q_MGR_NAME_LENGTH);
    MQCONN(QMName, &hConn, &compCode, &connReason);

/*****
/* Report reason and stop if connection failed */
/*****
if (compCode == MQCC_FAILED)
{
    CheckCallResult("MQCONN", compCode, connReason);
    exit( (int)connReason);
}

/*****
/* Call the routine to create a local queue, passing the handle to the
/* queue manager and also passing the name of the queue to be created.
/*
/*****
CreateLocalQueue(hConn, argv[1]);

/*****
/* Disconnect from the queue manager if not already connected */
/*****
if (connReason != MQRC_ALREADY_CONNECTED)
{
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("MQDISC", compCode, reason);
}
return 0;

}

/*****
/*
/* Function: CreateLocalQueue
/* Description: Create a local queue by sending a PCF command to the command
/* server.
/*
/*
/*****
/*
/* Input Parameters: Handle to the queue manager
/* Name of the queue to be created
/*
/*
/* Output Parameters: None
/*
/*
/* Logic: The mqExecute call is executed with the command MQCMD_CREATE_Q.
/* The call generates the correct PCF structure.
/* The default options to the call are used so that the command is sent
/* to the SYSTEM.ADMIN.COMMAND.QUEUE.
/* The reply from the command server is placed on a temporary dynamic
/* queue.
/* The reply is read from the temporary queue and formatted into the
/* response bag.
/*
/*
/* The completion code from the mqExecute call is checked and if there
/* is a failure from the command server then the code returned by the
/* command server is retrieved from the system bag that is
/* embedded in the response bag to the mqExecute call.
/*
/*
/*****
void CreateLocalQueue(MQHCONN hConn, MQCHAR *qName)
{
    MQLONG reason; /* reason code */
    MQLONG compCode; /* completion code */
    MQHBAG commandBag = MQHB_UNUSABLE_HBAG; /* command bag for mqExecute */
    MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute */
    MQHBAG resultBag; /* result bag from mqExecute */
    MQLONG mqExecuteCC; /* mqExecute completion code */
    MQLONG mqExecuteRC; /* mqExecute reason code */
}

```

```

printf("\nCreating Local Queue %s\n\n", qName);

/*****
/* Create a command Bag for the mqExecute call. Exit the function if the
/* create fails.
*****/
mqCreateBag(MQCBO_ADMIN_BAG, &commandBag, &compCode, &reason);
CheckCallResult("Create the command bag", compCode, reason);
if (compCode !=MQCC_OK)
    return;

/*****
/* Create a response Bag for the mqExecute call, exit the function if the
/* create fails.
*****/
mqCreateBag(MQCBO_ADMIN_BAG, &responseBag, &compCode, &reason);
CheckCallResult("Create the response bag", compCode, reason);
if (compCode !=MQCC_OK)
    return;

/*****
/* Put the name of the queue to be created into the command bag. This will
/* be used by the mqExecute call.
*****/
mqAddString(commandBag, MQCA_Q_NAME, MQBL_NULL_TERMINATED, qName, &compCode,
            &reason);
CheckCallResult("Add q name to command bag", compCode, reason);

/*****
/* Put queue type of local into the command bag. This will be used by the
/* mqExecute call.
*****/
mqAddInteger(commandBag, MQIA_Q_TYPE, MQQT_LOCAL, &compCode, &reason);
CheckCallResult("Add q type to command bag", compCode, reason);

/*****
/* Send the command to create the required local queue.
/* The mqExecute call will create the PCF structure required, send it to
/* the command server and receive the reply from the command server into
/* the response bag.
*****/
mqExecute(hConn, /* IBM MQ connection handle */
          MQCMD_CREATE_Q, /* Command to be executed */
          MQHB_NONE, /* No options bag */
          commandBag, /* Handle to bag containing commands */
          responseBag, /* Handle to bag to receive the response*/
          MQHO_NONE, /* Put msg on SYSTEM.ADMIN.COMMAND.QUEUE*/
          MQHO_NONE, /* Create a dynamic q for the response */
          &compCode, /* Completion code from the mqExecute */
          &reason); /* Reason code from mqExecute call */

if (reason == MQRCD_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <strmqcsv QMgrName>\n")
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("MQDISC", compCode, reason);
    exit(98);
}

/*****
/* Check the result from mqExecute call and find the error if it failed.
*****/
if ( compCode == MQCC_OK )
    printf("Local queue %s successfully created\n", qName);
else
{
    printf("Creation of local queue %s failed: Completion Code = %d
           qName, compCode, reason);
    if (reason == MQRCCF_COMMAND_FAILED)
    {
        /*****
        /* Get the system bag handle out of the mqExecute response bag.
        /* This bag contains the reason from the command server why the
        /* command failed.
        *****/
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &resultBag, &compCode,
                    &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****

```

```

    /* Get the completion code and reason code, returned by the command */
    /* server, from the embedded error bag. */
    /*******/
    mqInquireInteger(resultBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
        &compCode, &reason);
    CheckCallResult("Get the completion code from the result bag",
        compCode, reason);
    mqInquireInteger(resultBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
        &compCode, &reason);
    CheckCallResult("Get the reason code from the result bag", compCode,
        reason);
    printf("Error returned by the command server: Completion code = %d :
        Reason = %d\n", mqExecuteCC, mqExecuteRC);
}
}
/*******/
/* Delete the command bag if successfully created. */
/*******/
if (commandBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&commandBag, &compCode, &reason);
    CheckCallResult("Delete the command bag", compCode, reason);
}

/*******/
/* Delete the response bag if successfully created. */
/*******/
if (responseBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&responseBag, &compCode, &reason);
    CheckCallResult("Delete the response bag", compCode, reason);
}
} /* end of CreateLocalQueue */

/*******/
/*
/* Function: CheckCallResult */
/*
/*
/*******/
/*
/* Input Parameters: Description of call */
/* Completion code */
/* Reason code */
/*
/* Output Parameters: None */
/*
/* Logic: Display the description of the call, the completion code and the */
/* reason code if the completion code is not successful */
/*
/*
/*******/
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %d :
            Reason = %d\n", callText, cc, rc);
}
}

```

Exemple de programme C pour l'affichage d'événements à l'aide d'un moniteur d'événements (amqsaie.m.c)

L'exemple de programme C amqsaie.m.c illustre un moniteur d'événements de base à l'aide de MQAI.

```

/*******/
/*
/* Program name: AMQSAIEM.C */
/*
/* Description: Sample C program to demonstrate a basic event monitor */
/* using the IBM MQ Admin Interface (MQAI). */
/* Licensed Materials - Property of IBM */
/*
/* 63H9336 */
/* (c) Copyright IBM Corp. 1999, 2023. All Rights Reserved. */
/*
/* US Government Users Restricted Rights - Use, duplication or */
/* disclosure restricted by GSA ADP Schedule Contract with */
/*******/

```

```

/* IBM Corp. */
/*****
/*
/* Function: */
/* AMQSAIEM is a sample C program that demonstrates how to write a simple */
/* event monitor using the mqGetBag call and other MQAI calls. */
/*
/* The name of the event queue to be monitored is passed as a parameter */
/* to the program. This would usually be one of the system event queues:- */
/* SYSTEM.ADMIN.QMGR.EVENT Queue Manager events */
/* SYSTEM.ADMIN.PERFM.EVENT Performance events */
/* SYSTEM.ADMIN.CHANNEL.EVENT Channel events */
/* SYSTEM.ADMIN.LOGGER.EVENT Logger events */
/*
/* To monitor the queue manager event queue or the performance event queue, */
/* the attributes of the queue manager need to be changed to enable */
/* these events. For more information about this, see Part 1 of the */
/* Programmable System Management book. The queue manager attributes can */
/* be changed using either MQSC commands or the MQAI interface. */
/* Channel events are enabled by default. */
/*
/* Program logic */
/* Connect to the Queue Manager. */
/* Open the requested event queue with a wait interval of 30 seconds. */
/* Wait for a message, and when it arrives get the message from the queue */
/* and format it into an MQAI bag using the mqGetBag call. */
/* There are many types of event messages and it is beyond the scope of */
/* this sample to program for all event messages. Instead the program */
/* prints out the contents of the formatted bag. */
/* Loop around to wait for another message until either there is an error */
/* or the wait interval of 30 seconds is reached. */
/*
/*****
/*
/* AMQSAIEM has 2 parameters - the name of the event queue to be monitored */
/* - the queue manager name (optional) */
/*
/*****

/*****
/* Includes */
/*****
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

#include <cmqc.h> /* MQI */
#include <cmqcfc.h> /* PCF */
#include <cmqbc.h> /* MQAI */

/*****
/* Macros */
/*****
#if MQAT_DEFAULT == MQAT_WINDOWS_NT
#define Int64 "I64"
#elif defined(MQ_64_BIT)
#define Int64 "l"
#else
#define Int64 "ll"
#endif

/*****
/* Function prototypes */
/*****
void CheckCallResult(MQCHAR *, MQLONG , MQLONG);
void GetQEvents(MQHCONN, MQCHAR *);
int PrintBag(MQHBAG);
int PrintBagContents(MQHBAG, int);

/*****
/* Function: main */
/*****
int main(int argc, char *argv[])
{
    MQHCONN hConn; /* handle to connection */
    MQCHAR QMName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QM name */
    MQLONG reason; /* reason code */
    MQLONG connReason; /* MQCONN reason code */
    MQLONG compCode; /* completion code */

/*****

```

```

/* First check the required parameters */
/*****
printf("Sample Event Monitor (times out after 30 secs)\n");
if (argc < 2)
{
    printf("Required parameter missing - event queue to be monitored\n");
    exit(99);
}

/*****
/* Connect to the queue manager */
/*****
if (argc > 2)
    strncpy(QMName, argv[2], (size_t)MQ_Q_MGR_NAME_LENGTH);
MQCONN(QMName, &hConn, &compCode, &connReason);
/*****
/* Report the reason and stop if the connection failed */
/*****
if (compCode == MQCC_FAILED)
{
    CheckCallResult("MQCONN", compCode, connReason);
    exit( (int)connReason);
}

/*****
/* Call the routine to open the event queue and format any event messages */
/* read from the queue. */
/*****
GetQEvents(hConn, argv[1]);

/*****
/* Disconnect from the queue manager if not already connected */
/*****
if (connReason != MQRC_ALREADY_CONNECTED)
{
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("MQDISC", compCode, reason);
}

return 0;
}

/*****
/*
/* Function: CheckCallResult */
/*
/*****
/*
/* Input Parameters: Description of call */
/* Completion code */
/* Reason code */
/*
/* Output Parameters: None */
/*
/* Logic: Display the description of the call, the completion code and the */
/* reason code if the completion code is not successful */
/*
/*****
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %d : Reason = %d\n",
            callText, cc, rc);
}

/*****
/*
/* Function: GetQEvents */
/*
/*****
/*
/* Input Parameters: Handle to the queue manager */
/* Name of the event queue to be monitored */
/*
/* Output Parameters: None */
/*
/* Logic: Open the event queue. */
/* Get a message off the event queue and format the message into */
/* a bag. */
/* A real event monitor would need to be programmed to deal with */

```



```

/*      each type of event that it receives from the queue. This is      */
/*      outside the scope of this sample, so instead, the contents of    */
/*      the bag are printed.                                             */
/*      The program waits for 30 seconds for an event message and then   */
/*      terminates if no more messages are available.                   */
/*      */
/*****
void GetQEvents(MQHCONN hConn, MQCHAR *qName)
{
    MQLONG openReason;          /* MQOPEN reason code      */
    MQLONG reason;             /* reason code             */
    MQLONG compCode;          /* completion code        */
    MQHOBJ eventQueue;        /* handle to event queue   */

    MQHBAG eventBag = MQHB_UNUSABLE_HBAG; /* event bag to receive event msg */
    MQOD od = {MQOD_DEFAULT}; /* Object Descriptor      */
    MQMD md = {MQMD_DEFAULT}; /* Message Descriptor     */
    MQGMO gmo = {MQGMO_DEFAULT}; /* get message options   */
    MQLONG bQueueOK = 1; /* keep reading msgs while true */

    /*****
    /* Create an Event Bag in which to receive the event.                */
    /* Exit the function if the create fails.                             */
    /*****
    mqCreateBag(MQCBO_USER_BAG, &eventBag, &compCode, &reason);
    CheckCallResult("Create event bag", compCode, reason);
    if (compCode != MQCC_OK)
        return;

    /*****
    /* Open the event queue chosen by the user                            */
    /*****
    strncpy(od.ObjectName, qName, (size_t)MQ_Q_NAME_LENGTH);
    MQOPEN(hConn, &od, MQOO_INPUT_AS_Q_DEF+MQOO_FAIL_IF QUIESCING, &eventQueue,
            &compCode, &openReason);
    CheckCallResult("Open event queue", compCode, openReason);

    /*****
    /* Set the GMO options to control the action of the get message from the */
    /* queue.                                                                    */
    /*****
    gmo.WaitInterval = 30000; /* 30 second wait for message */
    gmo.Options = MQGMO_WAIT + MQGMO_FAIL_IF QUIESCING + MQGMO_CONVERT;
    gmo.Version = MQGMO_VERSION_2; /* Avoid need to reset Message ID */
    gmo.MatchOptions = MQMO_NONE; /* and Correlation ID after every */
                                /* mqGetBag

    /*****
    /* If open fails, we cannot access the queue and must stop the monitor. */
    /*****
    if (compCode != MQCC_OK)
        bQueueOK = 0;

    /*****
    /* Main loop to get an event message when it arrives                    */
    /*****
    while (bQueueOK)
    {
        printf("\nWaiting for an event\n");

        /*****
        /* Get the message from the event queue and convert it into the event */
        /* bag.                                                                    */
        /*****
        mqGetBag(hConn, eventQueue, &md, &gmo, eventBag, &compCode, &reason);

        /*****
        /* If get fails, we cannot access the queue and must stop the monitor. */
        /*****
        if (compCode != MQCC_OK)
        {
            bQueueOK = 0;

            /*****
            /* If get fails because no message available then we have timed out, */
            /* so report this, otherwise report an error.                            */
            /*****
            if (reason == MQRC_NO_MSG_AVAILABLE)
            {
                printf("No more messages\n");
            }
            else
            {

```

```

        CheckCallResult("Get bag", compCode, reason);
    }
}

/*****
/* Event message read - Print the contents of the event bag */
*****/
else
{
    if ( PrintBag(eventBag) )
        printf("\nError found while printing bag contents\n");

    } /* end of msg found */
} /* end of main loop */
/*****
/* Close the event queue if successfully opened */
*****/
if (openReason == MQRC_NONE)
{
    MQCLOSE(hConn, &eventQueue, MQCO_NONE, &compCode, &reason);
    CheckCallResult("Close event queue", compCode, reason);
}

/*****
/* Delete the event bag if successfully created. */
*****/
if (eventBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&eventBag, &compCode, &reason);
    CheckCallResult("Delete the event bag", compCode, reason);
}
} /* end of GetQEvents */

/*****
/*
/* Function: PrintBag
/*
*****/
/*
/* Input Parameters: Bag Handle
/*
/*
/* Output Parameters: None
/*
/*
/* Returns: Number of errors found
/*
/*
/* Logic: Calls PrintBagContents to display the contents of the bag.
/*
*****/

int PrintBag(MQHBAG dataBag)
{
    int errors;

    printf("\n");
    errors = PrintBagContents(dataBag, 0);
    printf("\n");

    return errors;
}

/*****
/*
/* Function: PrintBagContents
/*
*****/
/*
/* Input Parameters: Bag Handle
/*
/* Indentation level of bag
/*
/*
/* Output Parameters: None
/*
/*
/* Returns: Number of errors found
/*
/*
/* Logic: Count the number of items in the bag
/*
/* Obtain selector and item type for each item in the bag.
/*
/* Obtain the value of the item depending on item type and display the
/*
/* index of the item, the selector and the value.
/*
/* If the item is an embedded bag handle then call this function again
/*
/* to print the contents of the embedded bag increasing the
/*
/* indentation level.
*****/

```

```

/*****
int PrintBagContents(MQHBAG dataBag, int indent)
{
    /*****/
    /* Definitions */
    /*****/
    #define LENGTH 500          /* Max length of string to be read*/
    #define INDENT 4           /* Number of spaces to indent */
                                /* embedded bag display */

    /*****/
    /* Variables */
    /*****/
    MQLONG itemCount;          /* Number of items in the bag */
    MQLONG itemType;          /* Type of the item */
    int i;                     /* Index of item in the bag */
    MQCHAR stringVal[LENGTH+1]; /* Value if item is a string */
    MQBYTE byteStringVal[LENGTH]; /* Value if item is a byte string */
    MQLONG stringLength;      /* Length of string value */
    MQLONG ccsid;             /* CCSID of string value */
    MQINT32 iValue;           /* Value if item is an integer */
    MQINT64 i64Value;         /* Value if item is a 64-bit
                                /* integer */

    MQLONG selector;         /* Selector of item */
    MQHBAG bagHandle;        /* Value if item is a bag handle */
    MQLONG reason;           /* reason code */
    MQLONG compCode;         /* completion code */
    MQLONG trimLength;       /* Length of string to be trimmed */
    int errors = 0;          /* Count of errors found */
    char blanks[] = "        "; /* Blank string used to
                                /* indent display */

    /*****/
    /* Count the number of items in the bag */
    /*****/
    mqCountItems(dataBag, MQSEL_ALL_SELECTORS, &itemCount, &compCode, &reason);

    if (compCode != MQCC_OK)
        errors++;
    else
    {
        printf("
        printf("
        printf("
    }

    /*****/
    /* If no errors found, display each item in the bag */
    /*****/
    if (!errors)
    {
        for (i = 0; i < itemCount; i++)
        {
            /*****/
            /* First inquire the type of the item for each item in the bag */
            /*****/
            mqInquireItemInfo(dataBag,          /* Bag handle */
                              MQSEL_ANY_SELECTOR, /* Item can have any selector*/
                              i,                /* Index position in the bag */
                              &selector,        /* Actual value of selector */
                                                  /* returned by call */
                              &itemType,        /* Actual type of item */
                                                  /* returned by call */
                              &compCode,        /* Completion code */
                              &reason);        /* Reason Code */

            if (compCode != MQCC_OK)
                errors++;

            switch(itemType)
            {
            case MQITEM_INTEGER:
                /*****/
                /* Item is an integer. Find its value and display its index,
                /* selector and value. */
                /*****/
                mqInquireInteger(dataBag,      /* Bag handle */
                                 MQSEL_ANY_SELECTOR, /* Allow any selector */
                                 i,            /* Index position in the bag */
                                 &iValue,     /* Returned integer value

```

```

                                &compCode,      /* Completion code      */
                                &reason);        /* Reason Code          */

if (compCode != MQCC_OK)
    errors++;
else
    printf("%.s %-2d %-4d (%d)\n",
           indent, blanks, i, selector, iValue);
break

case MQITEM_INTEGER64:
/*****
/* Item is a 64-bit integer. Find its value and display its
/* index, selector and value.
*****/
mqInquireInteger64(dataBag, /* Bag handle
                    MQSEL_ANY_SELECTOR, /* Allow any selector
                    i, /* Index position in the bag
                    &i64Value, /* Returned integer value
                    &compCode, /* Completion code
                    &reason); /* Reason Code

if (compCode != MQCC_OK)
    errors++;
else
    printf("%.s %-2d %-4d (%"Int64"d)\n",
           indent, blanks, i, selector, i64Value);
break;

case MQITEM_STRING:
/*****
/* Item is a string. Obtain the string in a buffer, prepare
/* the string for displaying and display the index, selector,
/* string and Character Set ID.
*****/
mqInquireString(dataBag, /* Bag handle
                 MQSEL_ANY_SELECTOR, /* Allow any selector
                 i, /* Index position in the bag
                 LENGTH, /* Maximum length of buffer
                 stringVal, /* Buffer to receive string
                 &stringLength, /* Actual length of string
                 &ccsid, /* Coded character set ID
                 &compCode, /* Completion code
                 &reason); /* Reason Code

/*****
/* The call can return a warning if the string is too long for
/* the output buffer and has been truncated, so only check
/* explicitly for call failure.
*****/
if (compCode == MQCC_FAILED)
    errors++;
else
{
/*****
/* Remove trailing blanks from the string and terminate with
/* a null. First check that the string should not have been
/* longer than the maximum buffer size allowed.
*****/
if (stringLength > LENGTH)
    trimLength = LENGTH;
else
    trimLength = stringLength;
mqTrim(trimLength, stringVal, stringVal, &compCode, &reason);
printf("%.s %-2d %-4d '%s' %d\n",
       indent, blanks, i, selector, stringVal, ccsid);
}
break;

case MQITEM_BYTE_STRING:
/*****
/* Item is a byte string. Obtain the byte string in a buffer,
/* prepare the byte string for displaying and display the
/* index, selector and string.
*****/
mqInquireByteString(dataBag, /* Bag handle
                    MQSEL_ANY_SELECTOR, /* Allow any selector
                    i, /* Index position in the bag
                    LENGTH, /* Maximum length of buffer
                    byteStringVal, /* Buffer to receive string
                    &stringLength, /* Actual length of string

```

```

                                &compCode, /* Completion code */
                                &reason); /* Reason Code */

/*****
/* The call can return a warning if the string is too long for */
/* the output buffer and has been truncated, so only check */
/* explicitly for call failure. */
/*****
if (compCode == MQCC_FAILED)
    errors++;
else
{
    printf("%.s %-2d %-4d X'",
           indent, blanks, i, selector);

    for (i = 0 ; i < stringLength ; i++)
        printf("

    printf("\n");
}
break;

case MQITEM_BAG:
/*****
/* Item is an embedded bag handle, so call the PrintBagContents*/
/* function again to display the contents. */
/*****
mqInquireBag(dataBag, /* Bag handle */
             MQSEL_ANY_SELECTOR, /* Allow any selector */
             i, /* Index position in the bag */
             &bagHandle, /* Returned embedded bag hdl*/
             &compCode, /* Completion code */
             &reason); /* Reason Code */

if (compCode != MQCC_OK)
    errors++;
else
{
    printf("%.s %-2d %-4d (%d)\n", indent, blanks, i,
           selector, bagHandle);
    if (selector == MQHA_BAG_HANDLE)
        printf("
    else
        printf("
        PrintBagContents(bagHandle, indent+INDENT);
}
break;

default:
    printf("
}
}
}
return errors;
}

```

Exemple de programme C pour l'interrogation d'objets de canal (amqsaicl.c)

L'exemple de programme C amqsaicl.c demande des objets de canal à l'aide de MQAI.

```

/*****
/*
/* Program name: AMQSAICL.C
/*
/* Description: Sample C program to inquire channel objects
/* using the IBM MQ Administration Interface (MQAI)
/*
/*
/* <N_OCO_COPYRIGHT>
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 2008, 2023. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/* <NOC_COPYRIGHT>
/*****

```

```

/*                                                                    */
/* Function:                                                            */
/* AMQSAICL is a sample C program that demonstrates how to inquire    */
/* attributes of the local queue manager using the MQAI interface. In  */
/* particular, it inquires all channels and their types.              */
/*                                                                    */
/* - A PCF command is built from items placed into an MQAI administration */
/* bag.                                                                */
/* These are:-                                                         */
/*   - The generic channel name "*"                                  */
/*   - The attributes to be inquired. In this sample we just want    */
/*     name and type attributes                                     */
/*                                                                    */
/* - The mqExecute MQCMD_INQUIRE_CHANNEL call is executed.          */
/* The call generates the correct PCF structure.                      */
/* The default options to the call are used so that the command is sent */
/* to the SYSTEM.ADMIN.COMMAND.QUEUE.                                */
/* The reply from the command server is placed on a temporary dynamic  */
/* queue.                                                            */
/* The reply from the MQCMD_INQUIRE_CHANNEL is read from the        */
/* temporary queue and formatted into the response bag.              */
/*                                                                    */
/* - The completion code from the mqExecute call is checked and if there */
/* is a failure from the command server, then the code returned by the */
/* command server is retrieved from the system bag that has been    */
/* embedded in the response bag to the mqExecute call.              */
/*                                                                    */
/* Note: The command server must be running.                          */
/*                                                                    */
/*****                                                                    */
/* AMQSAICL has 2 parameter - the queue manager name (optional)      */
/*   - output file (optional) default varies                          */
/*****                                                                    */

/*****                                                                    */
/* Includes                                                            */
/*****                                                                    */
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>
#if (MQAT_DEFAULT == MQAT_OS400)
#include <recio.h>
#endif

#include <cmqc.h>                /* MQI                */
#include <cmqcfh.h>              /* PCF                */
#include <cmqbc.h>               /* MQAI               */
#include <cmqxc.h>              /* MQCD               */

/*****                                                                    */
/* Function prototypes                                                */
/*****                                                                    */
void CheckCallResult(MQCHAR *, MQLONG , MQLONG);

/*****                                                                    */
/* DataTypes                                                          */
/*****                                                                    */
#if (MQAT_DEFAULT == MQAT_OS400)
typedef _RFIL FILE;
#else
typedef FILE;
#endif

/*****                                                                    */
/* Constants                                                         */
/*****                                                                    */
#if (MQAT_DEFAULT == MQAT_OS400)
const struct
{
    char name[9];
} ChlTypeMap[9] =
{
    "*SDR      ", /* MQCHT_SENDER      */
    "*SVR      ", /* MQCHT_SERVER      */
    "*RCVR     ", /* MQCHT_RECEIVER   */
    "*RQSTR    ", /* MQCHT_REQUESTER  */
    "*ALL      ", /* MQCHT_ALL         */
    "*CLTCN    ", /* MQCHT_CLNTCONN   */
    "*SVRCONN  ", /* MQCHT_SVRCONN    */
    "*CLUSRCVR", /* MQCHT_CLUSRCVR   */
};

```

```

    "CLUSSDR "      /* MQCHT_CLUSSDR */
};
#else
const struct
{
    char name[9];
} ChlTypeMap[9] =
{
    "sdr      ", /* MQCHT_SENDER */
    "svr      ", /* MQCHT_SERVER */
    "rcvr     ", /* MQCHT_RECEIVER */
    "rqstr    ", /* MQCHT_REQUESTER */
    "all      ", /* MQCHT_ALL */
    "cltconn  ", /* MQCHT_CLNTCONN */
    "svrcn    ", /* MQCHT_SVRCONN */
    "clusrcvr ", /* MQCHT_CLUSRCVR */
    "cluscdr  ", /* MQCHT_CLUSSDR */
};
#endif

/*****
/* Macros */
/*****
#if (MQAT_DEFAULT == MQAT_OS400)
#define OUTFILE "QTEMP/AMQSAICL(AMQSAICL)"
#define OPENOUTFILE(hdl, fname) \
    (hdl) = _Ropen((fname), "wr", rtncode=Y);
#define CLOSEOUTFILE(hdl) \
    _Rclose((hdl));
#define WRITEOUTFILE(hdl, buf, buflen) \
    _Rwrite((hdl), (buf), (buflen));

#elif (MQAT_DEFAULT == MQAT_UNIX)
#define OUTFILE "/tmp/amqsaicl.txt"
#define OPENOUTFILE(hdl, fname) \
    (hdl) = fopen((fname), "w");
#define CLOSEOUTFILE(hdl) \
    fclose((hdl));
#define WRITEOUTFILE(hdl, buf, buflen) \
    fwrite((buf), (buflen), 1, (hdl)); fflush((hdl));

#else
#define OUTFILE "amqsaicl.txt"
#define OPENOUTFILE(fname) \
    fopen((fname), "w");
#define CLOSEOUTFILE(hdl) \
    fclose((hdl));
#define WRITEOUTFILE(hdl, buf, buflen) \
    fwrite((buf), (buflen), 1, (hdl)); fflush((hdl));

#endif

#define ChlType2String(t) ChlTypeMap[(t)-1].name

/*****
/* Function: main */
/*****
int main(int argc, char *argv[])
{
    /*****
    /* MQAI variables */
    /*****
    MQHCONN hConn; /* handle to MQ connection */
    MQCHAR qmName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name */
    MQLONG reason; /* reason code */
    MQLONG connReason; /* MQCONN reason code */
    MQLONG compCode; /* completion code */
    MQHBAG adminBag = MQHB_UNUSABLE_HBAG; /* admin bag for mqExecute */
    MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute */
    MQHBAG cAttrsBag; /* bag containing chl attributes */
    MQHBAG errorBag; /* bag containing cmd server error */
    MQLONG mqExecuteCC; /* mqExecute completion code */
    MQLONG mqExecuteRC; /* mqExecute reason code */
    MQLONG chlNameLength; /* Actual length of chl name */
    MQLONG chlType; /* Channel type */
    MQLONG i; /* loop counter */
    MQLONG numberOfBags; /* number of bags in response bag */
    MQCHAR chlName[MQ_OBJECT_NAME_LENGTH+1]; /* name of chl extracted from bag */
    MQCHAR OutputBuffer[100]; /* output data buffer */
    OUTFILEHDL *outfp = NULL; /* output file handle */

    /*****

```

```

/* Connect to the queue manager */
/*****
if (argc > 1)
    strncpy(qmName, argv[1], (size_t)MQ_Q_MGR_NAME_LENGTH);
MQCONN(qmName, &hConn;, &compCode;, &connReason;);

/*****
/* Report the reason and stop if the connection failed. */
/*****
if (compCode == MQCC_FAILED)
{
    CheckCallResult("Queue Manager connection", compCode, connReason);
    exit( (int)connReason);
}

/*****
/* Open the output file */
/*****
if (argc > 2)
{
    OPENOUTFILE(outfp, argv[2]);
}
else
{
    OPENOUTFILE(outfp, OUTFILE);
}

if(outfp == NULL)
{
    printf("Could not open output file.\n");
    goto MOD_EXIT;
}
/*****
/* Create an admin bag for the mqExecute call */
/*****
mqCreateBag(MQCBO_ADMIN_BAG, &adminBag;, &compCode;, &reason;);
CheckCallResult("Create admin bag", compCode, reason);

/*****
/* Create a response bag for the mqExecute call */
/*****
mqCreateBag(MQCBO_ADMIN_BAG, &responseBag;, &compCode;, &reason;);
CheckCallResult("Create response bag", compCode, reason);

/*****
/* Put the generic channel name into the admin bag */
/*****
mqAddString(adminBag, MQCACH_CHANNEL_NAME, MQBL_NULL_TERMINATED, "*",
            &compCode;, &reason;);
CheckCallResult("Add channel name", compCode, reason);

/*****
/* Put the channel type into the admin bag */
/*****
mqAddInteger(adminBag, MQIACH_CHANNEL_TYPE, MQCHT_ALL, &compCode;, &reason;);
CheckCallResult("Add channel type", compCode, reason);

/*****
/* Add an inquiry for various attributes */
/*****
mqAddInquiry(adminBag, MQIACH_CHANNEL_TYPE, &compCode;, &reason;);
CheckCallResult("Add inquiry", compCode, reason);

/*****
/* Send the command to find all the channel names and channel types. */
/* The mqExecute call creates the PCF structure required, sends it to */
/* the command server, and receives the reply from the command server into */
/* the response bag. The attributes are contained in system bags that are */
/* embedded in the response bag, one set of attributes per bag. */
/*****
mqExecute(hConn, /* MQ connection handle */
          MQCMD_INQUIRE_CHANNEL, /* Command to be executed */
          MQHB_NONE, /* No options bag */
          adminBag, /* Handle to bag containing commands */
          responseBag, /* Handle to bag to receive the response */
          MQHO_NONE, /* Put msg on SYSTEM.ADMIN.COMMAND.QUEUE */
          MQHO_NONE, /* Create a dynamic q for the response */
          &compCode;, /* Completion code from the mqexecute */
          &reason;); /* Reason code from mqexecute call */

/*****
/* Check the command server is started. If not exit. */
/*****

```



```

/*****
if (reason == MQRC_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <strmqcsv QMgrName=">\n");
    goto MOD_EXIT;
}

/*****
/* Check the result from mqExecute call. If successful find the channel */
/* types for all the channels. If failed find the error. */
/*****
if ( compCode == MQCC_OK ) /* Successful mqExecute */
{
    /*****
    /* Count the number of system bags embedded in the response bag from the */
    /* mqExecute call. The attributes for each channel are in separate bags. */
    /*****
mqCountItems(responseBag, MQHA_BAG_HANDLE, &numberOfBags,
              &compCode, &reason);
    CheckCallResult("Count number of bag handles", compCode, reason);

    for ( i=0; i<numberOfBags; i++)
    {
        /*****
        /* Get the next system bag handle out of the mqExecute response bag. */
        /* This bag contains the channel attributes */
        /*****
mqInquireBag(responseBag, MQHA_BAG_HANDLE, i, &cAttrsBag,
              &compCode, &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****
        /* Get the channel name out of the channel attributes bag */
        /*****
mqInquireString(cAttrsBag, MQCACH_CHANNEL_NAME, 0, MQ_OBJECT_NAME_LENGTH,
                 chlName, &chlNameLength, NULL, &compCode, &reason);
        CheckCallResult("Get channel name", compCode, reason);

        /*****
        /* Get the channel type out of the channel attributes bag */
        /*****

mqInquireInteger(cAttrsBag, MQIACH_CHANNEL_TYPE, MQIND_NONE, &chlType,
                 &compCode, &reason);
        CheckCallResult("Get type", compCode, reason);

        /*****
        /* Use mqTrim to prepare the channel name for printing. */
        /* Print the result. */
        /*****
mqTrim(MQ_CHANNEL_NAME_LENGTH, chlName, chlName, &compCode, &reason);
        sprintf(OutputBuffer, "%-20s%-9s", chlName, ChlType2String(chlType));
        WRITEOUTFILE(outfp, OutputBuffer, 29)
    }
}

else /* Failed mqExecute */
{
    printf("Call to get channel attributes failed: Cc = %ld : Rc = %ld\n",
           compCode, reason);
    /*****
    /* If the command fails get the system bag handle out of the mqexecute */
    /* response bag. This bag contains the reason from the command server */
    /* why the command failed. */
    /*****
if (reason == MQRCCF_COMMAND_FAILED)
{
    mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &errorBag,
                 &compCode, &reason);
    CheckCallResult("Get the result bag handle", compCode, reason);

    /*****
    /* Get the completion code and reason code, returned by the command */
    /* server, from the embedded error bag. */
    /*****
mqInquireInteger(errorBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
                 &compCode, &reason);
    CheckCallResult("Get the completion code from the result bag",
                    compCode, reason);
    mqInquireInteger(errorBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
                     &compCode, &reason);
    CheckCallResult("Get the reason code from the result bag",

```

```

        compCode, reason);
    printf("Error returned by the command server: Cc = %ld : Rc = %ld\n",
        mqExecuteCC, mqExecuteRC);
    }
}

MOD_EXIT:
/*****
/* Delete the admin bag if successfully created.
*****/
if (adminBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&adminBag, &compCode, &reason);
    CheckCallResult("Delete the admin bag", compCode, reason);
}

/*****
/* Delete the response bag if successfully created.
*****/
if (responseBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&responseBag, &compCode, &reason);
    CheckCallResult("Delete the response bag", compCode, reason);
}

/*****
/* Disconnect from the queue manager if not already connected
*****/
if (connReason != MQRC_ALREADY_CONNECTED)
{
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("Disconnect from Queue Manager", compCode, reason);
}

/*****
/* Close the output file if open
*****/
if(outfp != NULL)
    CLOSEOUTFILE(outfp);

return 0;
}

/*****
/*
/* Function: CheckCallResult
/*
*****/
/*
/* Input Parameters:  Description of call
/*                    Completion code
/*                    Reason code
/*
/* Output Parameters: None
/*
/* Logic: Display the description of the call, the completion code and the
/*        reason code if the completion code is not successful
/*
*****/
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %ld : Reason = %ld\n", callText,
            cc, rc);
}

```

Exemple de programme C pour l'interrogation des files d'attente et des informations d'impression (amqsailq.c)

L'exemple de programme C amqsailq.c demande la longueur en cours des files d'attente locales à l'aide de MQAI.

```

/*****
/*
/* Program name: AMQSAILQ.C
/*

```

```

/* Description: Sample C program to inquire the current depth of the local queues using the IBM MQ Administration Interface (MQAI)
/*
/*
/* Statement: Licensed Materials - Property of IBM
/*
/*
/*      84H2000, 5765-B73
/*      84H2001, 5639-B42
/*      84H2002, 5765-B74
/*      84H2003, 5765-B75
/*      84H2004, 5639-B43
/*
/*      (C) Copyright IBM Corp. 1999, 2023.
/*
/*
/*****
/*
/* Function:
/* AMQSAILQ is a sample C program that demonstrates how to inquire
/* attributes of the local queue manager using the MQAI interface. In
/* particular, it inquires the current depths of all the local queues.
/*
/* - A PCF command is built by placing items into an MQAI administration
/* bag.
/* These are:-
/* - The generic queue name "*"
/* - The type of queue required. In this sample we want to
/* inquire local queues.
/* - The attribute to be inquired. In this sample we want the
/* current depths.
/*
/* - The mqExecute call is executed with the command MQCMD_INQUIRE_Q.
/* The call generates the correct PCF structure.
/* The default options to the call are used so that the command is sent
/* to the SYSTEM.ADMIN.COMMAND.QUEUE.
/* The reply from the command server is placed on a temporary dynamic
/* queue.
/* The reply from the MQCMD_INQUIRE_Q command is read from the
/* temporary queue and formatted into the response bag.
/*
/* - The completion code from the mqExecute call is checked and if there
/* is a failure from the command server, then the code returned by
/* command server is retrieved from the system bag that has been
/* embedded in the response bag to the mqExecute call.
/*
/* - If the call is successful, the depth of each local queue is placed
/* in system bags embedded in the response bag of the mqExecute call.
/* The name and depth of each queue is obtained from each of the bags
/* and the result displayed on the screen.
/*
/* Note: The command server must be running.
/*
/*****
/*
/* AMQSAILQ has 1 parameter - the queue manager name (optional)
/*
/*****

/*****
/* Includes
/*****
#include <stdio.h>
#include <string.h>
#include <stdlib.h>
#include <ctype.h>

#include <cmqc.h>          /* MQI
#include <cmqcfc.h>       /* PCF
#include <cmqbc.h>        /* MQAI

/*****
/* Function prototypes
/*****
void CheckCallResult(MQCHAR *, MQLONG , MQLONG);

/*****
/* Function: main
/*****
int main(int argc, char *argv[])
{
    /*****
    /* MQAI variables
    /*****
    MQHCONN hConn;          /* handle to IBM MQ connection

```

```

MQCHAR qmName[MQ_Q_MGR_NAME_LENGTH+1]=""; /* default QMgr name */
MQLONG reason; /* reason code */
MQLONG connReason; /* MQCONN reason code */
MQLONG compCode; /* completion code */
MQHBAG adminBag = MQHB_UNUSABLE_HBAG; /* admin bag for mqExecute */
MQHBAG responseBag = MQHB_UNUSABLE_HBAG; /* response bag for mqExecute */
MQHBAG qAttrsBag; /* bag containing q attributes */
MQHBAG errorBag; /* bag containing cmd server error */
MQLONG mqExecuteCC; /* mqExecute completion code */
MQLONG mqExecuteRC; /* mqExecute reason code */
MQLONG qNameLength; /* Actual length of q name */
MQLONG qDepth; /* depth of queue */
MQLONG i; /* loop counter */
MQLONG numberOfBags; /* number of bags in response bag */
MQCHAR qName[MQ_Q_NAME_LENGTH+1]; /* name of queue extracted from bag*/

printf("Display current depths of local queues\n\n");

/*****
/* Connect to the queue manager */
/*****
if (argc > 1)
    strncpy(qmName, argv[1], (size_t)MQ_Q_MGR_NAME_LENGTH);
MQCONN(qmName, &hConn, &compCode, &connReason);

/*****
/* Report the reason and stop if the connection failed. */
/*****
if (compCode == MQCC_FAILED)
{
    CheckCallResult("Queue Manager connection", compCode, connReason);
    exit( (int)connReason);
}

/*****
/* Create an admin bag for the mqExecute call */
/*****
mqCreateBag(MQCBO_ADMIN_BAG, &adminBag, &compCode, &reason);
CheckCallResult("Create admin bag", compCode, reason);
/*****
/* Create a response bag for the mqExecute call */
/*****
mqCreateBag(MQCBO_ADMIN_BAG, &responseBag, &compCode, &reason);
CheckCallResult("Create response bag", compCode, reason);

/*****
/* Put the generic queue name into the admin bag */
/*****
mqAddString(adminBag, MQCA_Q_NAME, MQBL_NULL_TERMINATED, "*",
            &compCode, &reason);
CheckCallResult("Add q name", compCode, reason);

/*****
/* Put the local queue type into the admin bag */
/*****
mqAddInteger(adminBag, MQIA_Q_TYPE, MQQT_LOCAL, &compCode, &reason);
CheckCallResult("Add q type", compCode, reason);

/*****
/* Add an inquiry for current queue depths */
/*****
mqAddInquiry(adminBag, MQIA_CURRENT_Q_DEPTH, &compCode, &reason);
CheckCallResult("Add inquiry", compCode, reason);

/*****
/* Send the command to find all the local queue names and queue depths. */
/* The mqExecute call creates the PCF structure required, sends it to */
/* the command server, and receives the reply from the command server into */
/* the response bag. The attributes are contained in system bags that are */
/* embedded in the response bag, one set of attributes per bag. */
/*****
mqExecute(hConn, /* IBM MQ connection handle */
          MQCMD_INQUIRE_Q, /* Command to be executed */
          MQHB_NONE, /* No options bag */
          adminBag, /* Handle to bag containing commands */
          responseBag, /* Handle to bag to receive the response*/
          MQHO_NONE, /* Put msg on SYSTEM.ADMIN.COMMAND.QUEUE*/
          MQHO_NONE, /* Create a dynamic q for the response */
          &compCode, /* Completion code from the mqExecute */
          &reason); /* Reason code from mqExecute call */

```

```

/*****
/* Check the command server is started. If not exit. */
/*****
if (reason == MQRC_CMD_SERVER_NOT_AVAILABLE)
{
    printf("Please start the command server: <stimqcsv QMgrName>\n");
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("Disconnect from Queue Manager", compCode, reason);
    exit(98);
}

/*****
/* Check the result from mqExecute call. If successful find the current */
/* depths of all the local queues. If failed find the error. */
/*****
if ( compCode == MQCC_OK ) /* Successful mqExecute */
{
    /*****
    /* Count the number of system bags embedded in the response bag from the */
    /* mqExecute call. The attributes for each queue are in a separate bag. */
    /*****
    mqCountItems(responseBag, MQHA_BAG_HANDLE, &numberOfBags, &compCode,
        &reason);
    CheckCallResult("Count number of bag handles", compCode, reason);

    for ( i=0; i<numberOfBags; i++)
    {
        /*****
        /* Get the next system bag handle out of the mqExecute response bag. */
        /* This bag contains the queue attributes */
        /*****
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, i, &qAttrsBag, &compCode,
            &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****
        /* Get the queue name out of the queue attributes bag */
        /*****
        mqInquireString(qAttrsBag, MQCA_Q_NAME, 0, MQ_Q_NAME_LENGTH, qName,
            &qNameLength, NULL, &compCode, &reason);
        CheckCallResult("Get queue name", compCode, reason);

        /*****
        /* Get the depth out of the queue attributes bag */
        /*****
        mqInquireInteger(qAttrsBag, MQIA_CURRENT_Q_DEPTH, MQIND_NONE, &qDepth,
            &compCode, &reason);
        CheckCallResult("Get depth", compCode, reason);

        /*****
        /* Use mqTrim to prepare the queue name for printing. */
        /* Print the result. */
        /*****
        mqTrim(MQ_Q_NAME_LENGTH, qName, qName, &compCode, &reason)
        printf("%4d %-48s\n", qDepth, qName);
    }
}

else /* Failed mqExecute */
{
    printf("Call to get queue attributes failed: Completion Code = %d :
        Reason = %d\n", compCode, reason);

    /*****
    /* If the command fails get the system bag handle out of the mqExecute */
    /* response bag. This bag contains the reason from the command server */
    /* why the command failed. */
    /*****
    if (reason == MQRCCF_COMMAND_FAILED)
    {
        mqInquireBag(responseBag, MQHA_BAG_HANDLE, 0, &errorBag, &compCode,
            &reason);
        CheckCallResult("Get the result bag handle", compCode, reason);

        /*****
        /* Get the completion code and reason code, returned by the command */
        /* server, from the embedded error bag. */
        /*****
        mqInquireInteger(errorBag, MQIASY_COMP_CODE, MQIND_NONE, &mqExecuteCC,
            &compCode, &reason );
        CheckCallResult("Get the completion code from the result bag",

```

```

        compCode, reason);
mqInquireInteger(errorBag, MQIASY_REASON, MQIND_NONE, &mqExecuteRC,
                &compCode, &reason);
CheckCallResult("Get the reason code from the result bag",
                compCode, reason);
printf("Error returned by the command server: Completion Code = %d :
        Reason = %d\n", mqExecuteCC, mqExecuteRC);
    }
}

/*****
/* Delete the admin bag if successfully created.          */
*****/
if (adminBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&adminBag, &compCode, &reason);
    CheckCallResult("Delete the admin bag", compCode, reason);
}

/*****
/* Delete the response bag if successfully created.      */
*****/
if (responseBag != MQHB_UNUSABLE_HBAG)
{
    mqDeleteBag(&responseBag, &compCode, &reason);
    CheckCallResult("Delete the response bag", compCode, reason);
}

/*****
/* Disconnect from the queue manager if not already connected */
*****/
if (connReason != MQRC_ALREADY_CONNECTED)
{
    MQDISC(&hConn, &compCode, &reason);
    CheckCallResult("Disconnect from queue manager", compCode, reason);
}
return 0;
}

*****/
* Function: CheckCallResult                               */
*                                                         */
* Input Parameters:  Description of call                 */
*                   Completion code                     */
*                   Reason code                         */
* Output Parameters: None                               */
* Logic: Display the description of the call, the completion code and the
*        reason code if the completion code is not successful
*
*****/
void CheckCallResult(char *callText, MQLONG cc, MQLONG rc)
{
    if (cc != MQCC_OK)
        printf("%s failed: Completion Code = %d : Reason = %d\n",
                callText, cc, rc);
}

```

Sacs de données et MQAI

Un sac de données est un moyen de gérer les propriétés ou les paramètres des objets à l'aide de l'interface d'administration IBM MQ (MQAI).

Sacs de données

- Le sac de données contient zéro ou plusieurs *éléments de données*. Ces éléments de données sont commandés dans le sac au fur et à mesure qu'ils sont placés dans le sac. Il s'agit de l' *ordre d'insertion*. Chaque élément de données contient un *sélecteur* qui identifie l'élément de données et une *valeur* de cet élément de données qui peut être un entier, un entier 64 bits, un filtre d'entier, une chaîne, un filtre de chaîne, une chaîne d'octets, un filtre de chaîne d'octets ou un descripteur d'un autre sac.

Les éléments de données sont décrits en détail dans [«Types d'élément de données disponibles dans MQAI»](#), à la page 65

Il existe deux types de sélecteur: *sélecteurs d'utilisateur* et *sélecteurs de système*. Ces éléments sont décrits dans [Sélecteurs MQAI](#). Les sélecteurs sont généralement uniques, mais il est possible d'avoir plusieurs valeurs pour le même sélecteur. Dans ce cas, un *index* identifie l'occurrence particulière du sélecteur requise. Les index sont décrits dans [«Indexation dans MQAI»](#), à la page 38.

Une hiérarchie de ces concepts est illustrée dans la [Figure 1](#).

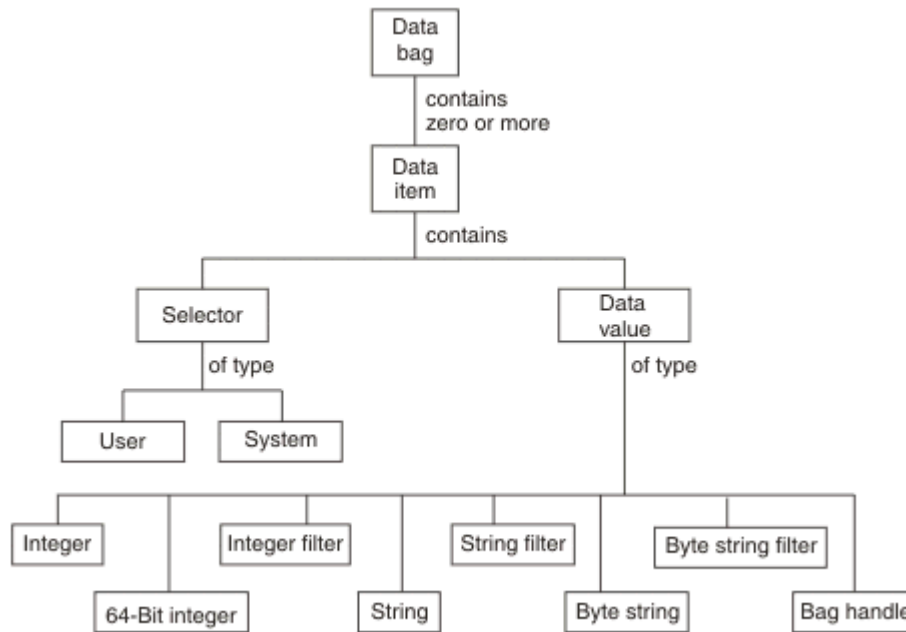


Figure 5. Hiérarchie des concepts MQAI

La hiérarchie a été expliquée dans un paragraphe précédent.

Types de sac de données

Vous pouvez choisir le type de sac de données que vous souhaitez créer en fonction de la tâche que vous souhaitez effectuer:

sac utilisateur

Sac simple utilisé pour les données utilisateur.

jeu de données d'administration

Sac créé pour les données utilisées pour administrer les objets IBM MQ en envoyant des messages d'administration à un serveur de commandes. Le sac d'administration implique automatiquement certaines options, comme décrit dans [«Création et suppression de sacs de données»](#), à la page 64.

sac de commandes

Un sac a également été créé pour les commandes d'administration des objets IBM MQ . Cependant, contrairement au sac d'administration, le sac de commande n'implique pas automatiquement certaines options bien que ces options soient disponibles. Pour plus d'informations sur les options, voir [«Création et suppression de sacs de données»](#), à la page 64.

sac de groupe

Sac utilisé pour contenir un ensemble d'éléments de données groupés. Les sacs de groupe ne peuvent pas être utilisés pour l'administration des objets IBM MQ .

En outre, le **sac système** est créé par MQAI lorsqu'un message de réponse est renvoyé par le serveur de commandes et placé dans le sac de sortie d'un utilisateur. Un sac système ne peut pas être modifié par l'utilisateur.

Utilisation des sacs de données Les différentes façons d'utiliser les sacs de données sont répertoriées dans cette rubrique:

Utilisation de sacs de données

Les différentes méthodes d'utilisation des sacs de données sont présentées dans la liste suivante:

- Vous pouvez créer et supprimer des sacs de données [«Création et suppression de sacs de données»](#), à la page 64.
- Vous pouvez envoyer des données entre des applications à l'aide de sacs de données [«Insertion et réception de sacs de données à l'aide de MQAI»](#), à la page 65.
- Vous pouvez ajouter des éléments de données à des sacs de données [«Ajout d'éléments de données à des sacs avec MQAI»](#), à la page 66.
- Vous pouvez ajouter une commande d'interrogation dans un sac de données [«Ajout d'une instruction d'interrogation à un sac»](#), à la page 67.
- Vous pouvez vous renseigner dans les sacs de données [«Recherche dans les sacs de données»](#), à la page 67.
- Vous pouvez compter des éléments de données dans un sac de données [«Comptage des éléments de données»](#), à la page 70.
- Vous pouvez modifier des informations dans un sac de données [«Modification d'informations dans un sac»](#), à la page 68.
- Vous pouvez effacer un sac de données [«Effacement d'un sac à l'aide de l'appel de sac mqClear»](#), à la page 69.
- Vous pouvez tronquer un sac de données [«Troncation d'un sac à l'aide de l'appel de sac mqTruncate»](#), à la page 69.
- Vous pouvez convertir des sacs et des tampons [«Conversion de sacs et de tampons»](#), à la page 69.

Création et suppression de sacs de données

Création de sacs de données

Pour utiliser MQAI, vous devez d'abord créer un sac de données à l'aide de l'appel mqCreateBag. En entrée de cet appel, vous fournissez une ou plusieurs options pour contrôler la création du sac.

Le paramètre **Options** de l'appel MQCreateBag vous permet de choisir de créer un sac d'utilisateur, un sac de commande, un sac de groupe ou un sac d'administration.

Pour créer un sac d'utilisateur, un sac de commande ou un sac de groupe, vous pouvez choisir une ou plusieurs options supplémentaires pour:

- Utilisez le formulaire de liste lorsqu'il y a plusieurs occurrences adjacentes du même sélecteur dans un sac.
- Réorganisez les éléments de données au fur et à mesure qu'ils sont ajoutés à un message PCF pour vous assurer que les paramètres sont dans le bon ordre. Pour plus d'informations sur les éléments de données, voir [«Types d'élément de données disponibles dans MQAI»](#), à la page 65.
- Vérifiez les valeurs des sélecteurs d'utilisateur pour les éléments que vous ajoutez au sac.

Les sacs d'administration impliquent automatiquement ces options.

Un sac de données est identifié par sa poignée. Le descripteur de sac est renvoyé par le sac mqCreateet doit être fourni sur tous les autres appels qui utilisent le sac de données.

Pour une description complète de l'appel de sac mqCreate, voir [mqCreateBag](#).

Suppression de sacs de données

Tout sac de données créé par l'utilisateur doit également être supprimé à l'aide de l'appel de sac `mqDelete`. Par exemple, si un sac est créé dans le code utilisateur, il doit également être supprimé dans le code utilisateur.

Les sacs système sont créés et supprimés automatiquement par MQAI. Pour plus d'informations à ce sujet, voir «[Envoi de commandes d'administration au serveur de commandes qm à l'aide de l'appel `mqExecute`](#)», à la page 71. Le code utilisateur ne peut pas supprimer un sac système.

Pour une description complète de l'appel de sac `mqDelete`, voir [mqDeleteBag](#).

Insertion et réception de sacs de données à l'aide de MQAI

Les données peuvent également être envoyées entre les applications en plaçant et en obtenant des sacs de données à l'aide des appels de sac `mqPut` et `mqGet`. Cela permet à l'interface d'administration IBM MQ (MQAI) de gérer la mémoire tampon plutôt que l'application.

L'appel de sac `mqPut` convertit le contenu du sac spécifié en message PCF et envoie le message à la file d'attente spécifiée ; l'appel de sac `mqGet` supprime le message de la file d'attente spécifiée et le reconvertit en sac de données. Par conséquent, l'appel `mqPut` est l'équivalent de l'appel `mqBagToBuffer` suivi de `MQPUT`, et l'appel `mqGet` est l'équivalent de l'appel `MQGET` suivi de `mqBufferToBag`.

Pour plus d'informations sur l'envoi et la réception de messages PCF dans une file d'attente spécifique, voir «[Envoi et réception de messages PCF dans une file d'attente spécifiée](#)», à la page 25

Remarque : Si vous choisissez d'utiliser l'appel `mqGet`, les détails PCF dans le message doivent être corrects ; si ce n'est pas le cas, une erreur appropriée se produit et le message PCF n'est pas renvoyé.

Types d'élément de données disponibles dans MQAI

Les éléments de données sont utilisés par l'interface d'administration IBM MQ (MQAI) pour remplir les sacs de données lorsqu'ils sont créés. Ces éléments de données peuvent être des éléments utilisateur ou système.

Ces éléments utilisateur contiennent des données utilisateur telles que les attributs des objets en cours de gestion. Les éléments système doivent être utilisés pour mieux contrôler les messages générés: par exemple, la génération d'en-têtes de message. Pour plus d'informations sur les éléments système, voir «[Éléments système et MQAI](#)», à la page 66.

Types d'éléments de données

Une fois que vous avez créé un sac de données, vous pouvez le remplir avec des éléments de type entier ou chaîne de caractères. Vous pouvez vous renseigner sur les trois types d'élément.

L'élément de données peut être un entier ou une chaîne de caractères. Voici les types d'élément de données disponibles dans MQAI:

- Entier
- Entier 64 bits
- Filtre de type entier
- Chaîne de caractères
- Filtre de chaîne
- Chaîne d'octets
- Filtre de chaîne d'octets
- Poignée de sac

Utilisation d'éléments de données

Les méthodes d'utilisation des éléments de données sont les suivantes:

- «[Comptage des éléments de données](#)», à la page 70.
- «[Suppression d'éléments de données](#)», à la page 70.

- «Ajout d'éléments de données à des sacs avec MQAI», à la page 66.
- «Filtrage et interrogation des éléments de données», à la page 67.

Eléments système et MQAI

Les éléments système peuvent être utilisés par l'interface d'administration IBM MQ (MQAI) pour:

- Génération des en-têtes PCF. Les éléments système peuvent contrôler l'identificateur de commande PCF, les options de contrôle, le numéro de séquence de message et le type de commande.
- Conversion de données. Les éléments système gèrent l'identificateur de jeu de caractères pour les éléments de chaîne de caractères du sac.

Comme tous les éléments de données, les éléments système sont constitués d'un sélecteur et d'une valeur. Pour plus d'informations sur ces sélecteurs et sur leur fonction, voir [Sélecteurs MQAI](#).

Les éléments système sont uniques. Un ou plusieurs éléments système peuvent être identifiés par un sélecteur de système. Il n'y a qu'une seule occurrence de chaque sélecteur de système.

La plupart des éléments système peuvent être modifiés (voir «[Modification d'informations dans un sac](#)», à la page 68), mais les options de création de sac ne peuvent pas être modifiées par l'utilisateur. Vous ne pouvez pas supprimer d'éléments système. (Voir «[Suppression d'éléments de données](#)», à la page 70.)

Ajout d'éléments de données à des sacs avec MQAI

Lorsqu'un sac de données est créé à l'aide de l'interface d'administration IBM MQ (MQAI), vous pouvez le remplir avec des éléments de données. Ces éléments de données peuvent être des éléments utilisateur ou système.

Pour plus d'informations sur les éléments de données, voir «[Types d'élément de données disponibles dans MQAI](#)», à la page 65.

L'interface MQAI vous permet d'ajouter des éléments de type entier, des éléments de type entier 64 bits, des éléments de type filtre de type entier, des éléments de type chaîne de caractères, des éléments de type chaîne de caractères, des éléments de type chaîne d'octets et des éléments de type filtre de type chaîne d'octets à des sacs, comme indiqué dans la [Figure 6](#), à la page 66. Les éléments sont identifiés par un sélecteur. Généralement, un sélecteur identifie un seul élément, mais ce n'est pas toujours le cas. Si un élément de données avec le sélecteur spécifié est déjà présent dans le sac, une instance supplémentaire de ce sélecteur est ajoutée à la fin du sac.



Figure 6. Ajout d'éléments de données

Ajoutez des éléments de données à un sac à l'aide des appels mqAdd*:

- Pour ajouter des éléments de type entier, utilisez l'appel mqAddInteger comme décrit dans [mqAddInteger](#)
- Pour ajouter des entiers 64 bits, utilisez l'appel mqAddInteger64 comme décrit dans [mqAddInteger64](#)
- Pour ajouter des éléments de filtre de type entier, utilisez l'appel mqAddIntegerFilter comme décrit dans [mqAddIntegerFilter](#)
- Pour ajouter des éléments de chaîne de caractères, utilisez l'appel de chaîne mqAddcomme décrit dans [mqAddString](#)

- Pour ajouter des éléments de filtre de chaîne, utilisez l'appel `mqAddStringFilter` comme décrit dans [mqAddStringFilter](#)
- Pour ajouter des éléments de chaîne d'octets, utilisez l'appel `mqAddByteString` comme décrit dans [mqAddByteString](#)
- Pour ajouter des éléments de filtre de chaîne d'octets, utilisez l'appel de filtre `mqAddByteString` comme décrit dans [mqAddByteStringFilter](#)

Pour plus d'informations sur l'ajout d'éléments de données à un sac, voir «[Eléments système et MQAI](#)», à la page 66.

Ajout d'une instruction d'interrogation à un sac

L'appel d'interrogation `mqAdd` permet d'ajouter une commande d'interrogation à un sac. L'appel est spécifique à des fins d'administration, de sorte qu'il peut être utilisé avec des sacs d'administration uniquement. Il vous permet de spécifier les sélecteurs d'attributs sur lesquels vous souhaitez vous renseigner à partir de IBM MQ.

Pour une description complète de l'appel `mqAddInquiry`, voir [mqAddInquiry](#).

Filtrage et interrogation des éléments de données

Lorsque vous utilisez MQAI pour vous renseigner sur les attributs des objets IBM MQ, vous pouvez contrôler les données renvoyées à votre programme de deux manières.

- Vous pouvez **filtrer** les données renvoyées à l'aide des appels de chaîne `mqAddInteger` et `mqAdd`. Cette approche vous permet de spécifier une paire *Selector* et *ItemValue*, par exemple:

```
mqAddInteger(inputbag, MQIA_Q_TYPE, MQQT_LOCAL)
```

Cet exemple indique que le type de file d'attente (*Selector*) doit être local (*ItemValue*) et que cette spécification doit correspondre aux attributs de l'objet (dans ce cas, une file d'attente) que vous interrogez.

Les autres attributs qui peuvent être filtrés correspondent aux commandes PCF Inquire * qui se trouvent dans «[Introduction à IBM MQ Programmable Command Formats](#)», à la page 22. Par exemple, pour en savoir plus sur les attributs d'un canal, reportez-vous à la commande Inquire Channel dans la documentation du produit. Les "paramètres requis" et les "paramètres facultatifs" de la commande Inquire Channel identifient les sélecteurs que vous pouvez utiliser pour le filtrage.

- Vous pouvez **interroger** les attributs particuliers d'un objet à l'aide de l'appel d'interrogation `mqAdd`. Indique le sélecteur qui vous intéresse. Si vous ne spécifiez pas le sélecteur, tous les attributs de l'objet sont renvoyés.

Voici un exemple de filtrage et d'interrogation des attributs d'une file d'attente:

```
/* Request information about all queues */
mqAddString(adminbag, MQCA_Q_NAME, "*")

/* Filter attributes so that local queues only are returned */
mqAddInteger(adminbag, MQIA_Q_TYPE, MQQT_LOCAL)

/* Query the names and current depths of the local queues */
mqAddInquiry(adminbag, MQCA_Q_NAME)
mqAddInquiry(adminbag, MQIA_CURRENT_Q_DEPTH)

/* Send inquiry to the command server and wait for reply */
mqExecute(MQCMD_INQUIRE_Q, ...)
```

Recherche dans les sacs de données

Vous pouvez vous renseigner sur:

- Valeur d'un élément entier à l'aide de l'appel `mqInquireInteger`. Voir [mqInquireInteger](#).
- Valeur d'un élément entier 64 bits à l'aide de l'appel `mqInquireInteger64`. Voir [mqInquireInteger64](#).

- Valeur d'un élément de filtre d'entier à l'aide de l'appel `mqInquireIntegerFilter` . Voir [mqInquireIntegerFilter](#).
- Valeur d'un élément de chaîne de caractères à l'aide de l'appel de chaîne `mqInquire`. Voir [mqInquireString](#).
- Valeur d'un élément de filtre de chaîne à l'aide de l'appel `mqInquireStringFilter` `mqInquire`. Voir [mqInquireStringFilter](#).
- Valeur d'un élément de chaîne d'octets à l'aide de l'appel `mqInquireByteString` . Voir [mqInquireByteString](#).
- Valeur d'un élément de filtre de chaîne d'octet à l'aide de l'appel de filtre `mqInquireByteString`. Voir [mqInquireByteStringFilter](#).
- Valeur d'un descripteur de sac à l'aide de l'appel de sac `mqInquire`. Voir [mqInquireBag](#).

Vous pouvez également vous renseigner sur le type (entier, entier 64 bits, filtre entier, chaîne de caractères, filtre de chaîne, chaîne d'octets, filtre de chaîne d'octets ou descripteur de sac) d'un élément spécifique à l'aide de l'appel `mqInquireItemInfo` . Voir [mqInquireItemInfo](#).

Modification d'informations dans un sac

L'interface MQAI vous permet de modifier des informations dans un sac à l'aide des appels `mqSet*`. Vous pouvez :

1. Modifier des éléments de données dans un sac. L'index permet de remplacer une instance individuelle d'un paramètre en identifiant l'occurrence de l'élément à modifier (voir [Figure 7](#), à la page 68).

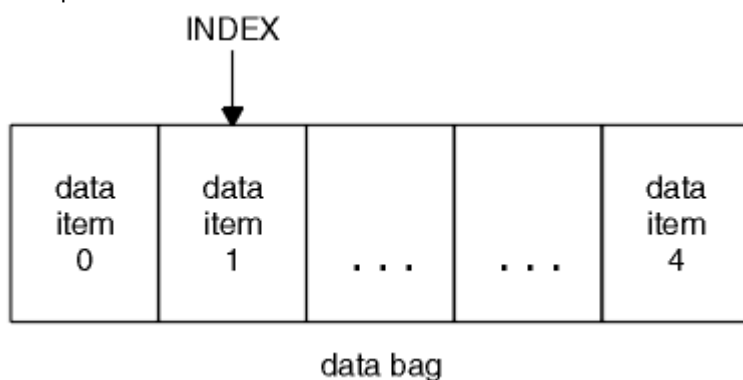


Figure 7. Modification d'un élément de données unique

2. Supprimez toutes les occurrences existantes du sélecteur spécifié et ajoutez une nouvelle occurrence à la fin du sac. (Voir [Figure 8](#), à la page 68.) Une valeur d'index spéciale permet de remplacer **toutes les instances** d'un paramètre.

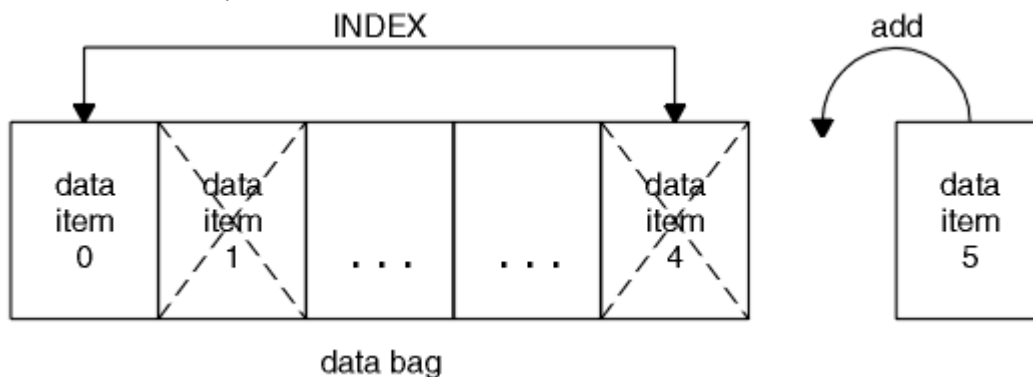


Figure 8. Modification de tous les éléments de données

Remarque : L'index conserve l'ordre d'insertion dans le sac mais peut affecter les index d'autres éléments de données.

L'appel `mqSetInteger` vous permet de modifier des éléments de type entier dans un sac. L'appel `mqSetInteger64` vous permet de modifier des éléments d'entier 64 bits. L'appel `mqSetIntegerFilter` vous permet de modifier des éléments de filtre de type entier. L'appel de chaîne `mqSet` vous permet de modifier des éléments de chaîne de caractères. L'appel `mqSetStringFilter` vous permet de modifier des éléments de filtre de chaîne. L'appel `mqSetByteString` vous permet de modifier des éléments de chaîne d'octets. L'appel de filtre `mqSetByteString` vous permet de modifier des éléments de filtre de chaîne d'octets. Vous pouvez également utiliser ces appels pour supprimer toutes les occurrences existantes du sélecteur spécifié et ajouter une nouvelle occurrence à la fin du sac. L'élément de données peut être un élément utilisateur ou un élément système.

Pour une description complète de ces appels, voir:

- [mqSetentier](#)
- [mqSetInteger64](#)
- [mqSetIntegerFilter](#)
- [mqSetChaîne](#)
- [mqSetStringFilter](#)
- [mqSetByteString](#)
- [mqSetByteStringFiltre](#)

Effacement d'un sac à l'aide de l'appel de sac `mqClear`

L'appel de sac `mqClear` supprime tous les éléments utilisateur d'un sac utilisateur et réinitialise les éléments système à leurs valeurs initiales. Les sacs système contenus dans le sac sont également supprimés.

Pour une description complète de l'appel de sac `mqClear`, voir [mqClearBag](#).

Troncation d'un sac à l'aide de l'appel de sac `mqTruncate`

L'appel de sac `mqTruncate` réduit le nombre d'éléments utilisateur dans un sac utilisateur en supprimant les éléments de la fin du sac, en commençant par l'élément ajouté le plus récemment. Par exemple, il peut être utilisé lors de l'utilisation des mêmes informations d'en-tête pour générer plusieurs messages.

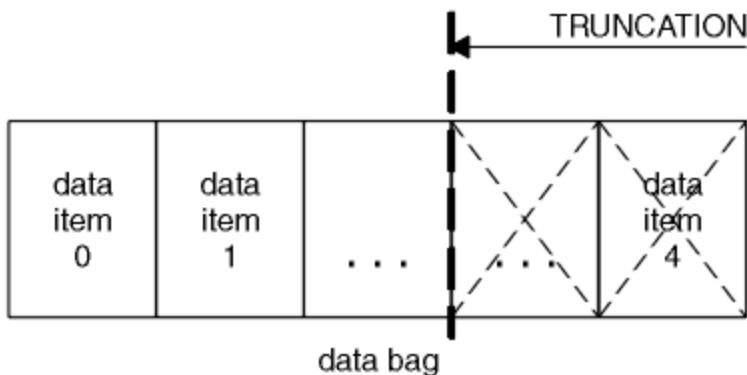


Figure 9. Troncation d'un sac

Pour une description complète de l'appel de sac `mqTruncate`, voir [mqTruncateBag](#).

Conversion de sacs et de tampons

Pour envoyer des données entre applications, on place d'abord les données de message dans un sac. Ensuite, les données du sac sont converties en message PCF à l'aide de l'appel `mqBagToBuffer`. Le message PCF est envoyé à la file d'attente requise à l'aide de l'appel `MQPUT`. Cela est illustré dans la figure Figure 10, à la page 70. Pour une description complète de l'appel `mqBagToBuffer`, voir [mqBagToBuffer](#).

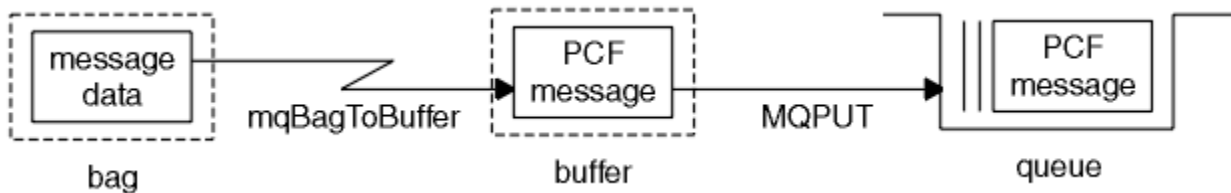


Figure 10. Conversion de sacs en messages PCF

Pour recevoir des données, le message est reçu dans une mémoire tampon à l'aide de l'appel MQGET. Les données de la mémoire tampon sont ensuite converties en un sac à l'aide de l'appel `mqBufferToBag`, à condition que la mémoire tampon contienne un message PCF valide. Cela est illustré dans la figure Figure 11, à la page 70. Pour une description complète de l'appel `mqBufferToBag`, voir `mqBufferToBag`.

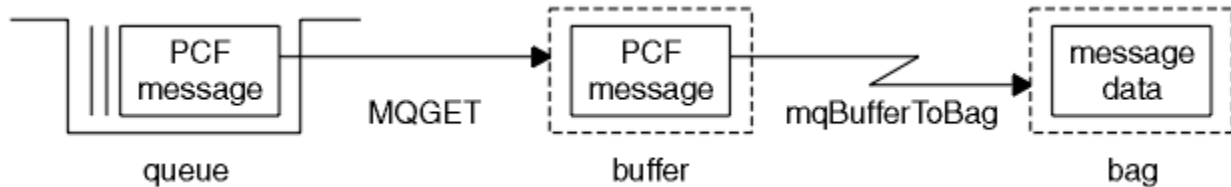


Figure 11. Conversion de messages PCF en forme de sac

Comptage des éléments de données

L'appel d'éléments `mqCount` compte le nombre d'éléments utilisateur, d'éléments système, ou les deux, qui sont stockés dans un sac de données et renvoie ce nombre. Par exemple, `mqCountItems (Bag , 7 , ...)` renvoie le nombre d'éléments dans le sac avec un sélecteur de 7. Il peut compter des éléments par sélecteur individuel, par sélecteur d'utilisateur, par sélecteur système ou par tous les sélecteurs.

Remarque : Cet appel compte le nombre d'éléments de données, et non le nombre de sélecteurs uniques dans le sac. Un sélecteur peut se produire plusieurs fois, de sorte qu'il peut y avoir moins de sélecteurs uniques dans le sac que d'éléments de données.

Pour une description complète de l'appel `mqCountItems`, voir `mqCountItems`.

Suppression d'éléments de données

Vous pouvez supprimer des éléments de sacs de plusieurs manières. Vous pouvez :

- Retirez un ou plusieurs éléments utilisateur d'un sac. Pour des informations détaillées, voir «Suppression d'éléments de données d'un sac à l'aide de l'appel d'élément `mqDelete`», à la page 70.
- Supprimez **tous** les éléments utilisateur d'un sac, c'est-à-dire *désélectionnez* un sac. Pour plus d'informations, voir «Effacement d'un sac à l'aide de l'appel de sac `mqClear`», à la page 69.
- Supprimer des éléments utilisateur de la fin d'un sac, c'est-à-dire *tronquer* un sac. Pour des informations détaillées, voir «Troncation d'un sac à l'aide de l'appel de sac `mqTruncate`», à la page 69.

Suppression d'éléments de données d'un sac à l'aide de l'appel d'élément `mqDelete`

L'appel d'élément `mqDelete` supprime un ou plusieurs éléments utilisateur d'un sac. L'index est utilisé pour supprimer:

1. Une seule occurrence du sélecteur spécifié. (Voir Figure 12, à la page 71.)

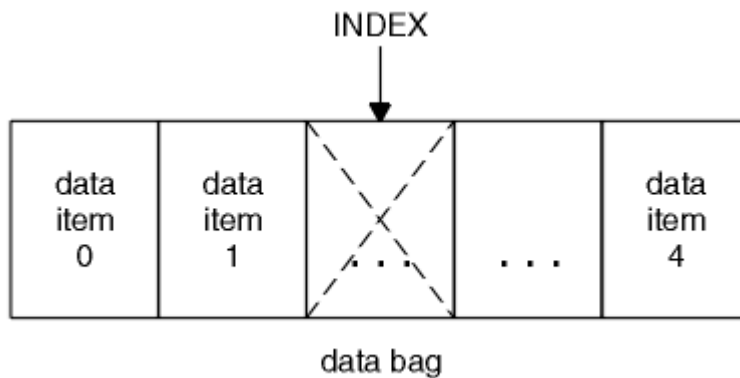


Figure 12. Suppression d'un élément de données unique

ou

2. Toutes les occurrences du sélecteur spécifié. (Voir [Figure 13](#), à la page 71.)

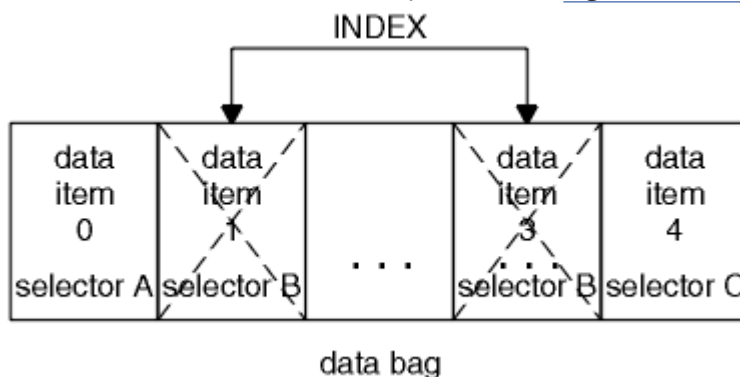


Figure 13. Suppression de tous les éléments de données

Remarque : L'index conserve l'ordre d'insertion dans le sac mais peut affecter les index d'autres éléments de données. Par exemple, l'appel d'élément `mqDelete` ne conserve pas les valeurs d'index des éléments de données qui suivent l'élément supprimé car les index sont réorganisés pour combler l'écart qui reste de l'élément supprimé.

Pour une description complète de l'appel d'élément `mqDelete`, voir [mqDeleteItem](#).

Envoi de commandes d'administration au serveur de commandes qm à l'aide de l'appel `mqExecute`

Lorsqu'un sac de données a été créé et rempli, un message de commande d'administration peut être envoyé au serveur de commandes d'un gestionnaire de files d'attente à l'aide de l'appel `mqExecute`. Cela permet de gérer l'échange avec le serveur de commandes et de renvoyer les réponses dans un sac.

Une fois que vous avez créé et rempli votre sac de données, vous pouvez envoyer un message de commande d'administration au serveur de commandes d'un gestionnaire de files d'attente. La méthode la plus simple consiste à utiliser l'appel `mqExecute`. L'appel `mqExecute` envoie un message de commande d'administration en tant que message non persistant et attend les réponses. Les réponses sont renvoyées dans un sac de réponse. Ils peuvent contenir des informations sur les attributs relatifs à plusieurs objets IBM MQ ou à une série de messages de réponse d'erreur PCF, par exemple. Par conséquent, le sac de réponse peut contenir un code retour uniquement ou *des sacs imbriqués*.

Les messages de réponse sont placés dans des sacs système créés par le système. Par exemple, pour les demandes concernant les noms d'objets, un sac système est créé pour contenir ces noms d'objet et le sac est inséré dans le sac utilisateur. Les poignées de ces sacs sont ensuite insérées dans le sac de réponse et le sac imbriqué est accessible par le sélecteur `MQHA_BAG_HANDLE`. Le sac système reste en stockage, s'il n'est pas supprimé, jusqu'à ce que le sac de réponse soit supprimé.

Le concept d' *imbrication* est illustré dans la [Figure 14](#), à la page 72.

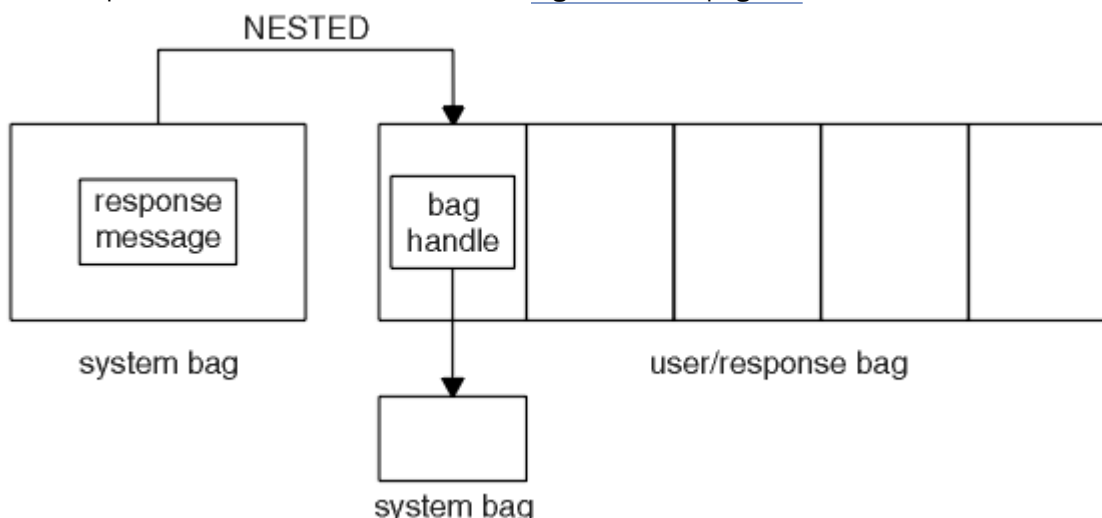


Figure 14. Imbrication

En tant qu'entrée de l'appel mqExecute , vous devez fournir:

- Un descripteur de connexion MQI.
- Commande à exécuter. Il doit s'agir de l'une des valeurs MQCMD_ *.

Remarque : Si cette valeur n'est pas reconnue par MQAI, elle est toujours acceptée. Toutefois, si l'appel d'interrogation mqAdda été utilisé pour insérer des valeurs dans le sac, ce paramètre doit être une commande INQUIRE reconnue par MQAI. Autrement dit, le paramètre doit être de la forme MQCMD_INQUIRE_ *.

- Eventuellement, une poignée du sac contenant des options qui contrôlent le traitement de l'appel. C'est également là que vous pouvez spécifier la durée maximale en millisecondes pendant laquelle MQAI doit attendre chaque message de réponse.
- Descripteur du sac d'administration qui contient les détails de la commande d'administration à émettre.
- Descripteur du sac de réponse qui reçoit les messages de réponse.

Les descripteurs suivants sont facultatifs:

- Descripteur d'objet de la file d'attente dans laquelle la commande d'administration doit être placée.
Si aucun descripteur d'objet n'est spécifié, la commande d'administration est placée sur SYSTEM.ADMIN.COMMAND.QUEUE appartenant au gestionnaire de files d'attente actuellement connecté. Il s'agit de l'option par défaut.
- Descripteur d'objet de la file d'attente dans laquelle les messages de réponse doivent être placés.

Vous pouvez choisir de placer les messages de réponse dans une file d'attente dynamique créée automatiquement par MQAI. La file d'attente créée existe uniquement pour la durée de l'appel et est supprimée par l'interface MQAI à la sortie de l'appel mqExecute .

Pour obtenir des exemples d'utilisation de l'appel mqExecute , voir [Exemple de code](#)

V 9.0.1 Administration à l'aide de REST API

Vous pouvez utiliser l' administrative REST API pour administrer les objets IBM MQ , tels que les gestionnaires de files d'attente et les files d'attente [V 9.0.5](#) , et les Managed File Transfer services REST GET. Les informations sont envoyées au administrative REST API et reçues de ce dernier au format JSON. Ces API RESTful peuvent vous aider à intégrer l'administration IBM MQ dans des DevOps et des outils d'automatisation populaires.

Avant de commencer

Pour des informations de référence sur les ressources REST disponibles, voir [The administrative REST API reference](#).

Remarque :

Dans IBM MQ 9.0.1, administrative REST API n'est pas intégré à la sécurité IBM MQ . Par conséquent, administrative REST API est désactivé par défaut. Vous devez activer manuellement le administrative REST API avant de pouvoir l'utiliser. Pour plus d'informations sur l'activation du administrative REST API, voir «[Guide d'initiation à administrative REST API](#)», à la page 73.

V 9.0.2 Depuis la IBM MQ 9.0.2, administrative REST API est intégré à la sécurité IBM MQ . administrative REST API est activé par défaut. Toutefois, vous devez configurer la sécurité pour pouvoir utiliser le administrative REST API. Pour plus d'informations, voir [IBM MQ Console et sécurité de l'REST API](#).

Procédure

- «[Guide d'initiation à administrative REST API](#)», à la page 73
- «[Utilisation de administrative REST API](#)», à la page 79
- «[Administration à distance à l'aide de REST API](#)», à la page 85
- «[REST API Horodatages](#)», à la page 89
- «[REST API traitement des erreurs](#)», à la page 89
- «[REST API découverte](#)», à la page 92
- «[REST APISupport de langue nationale \(NLS\)](#)», à la page 93

V 9.0.1 Guide d'initiation à administrative REST API

Avant de démarrer le administrative REST API , vous devez installer les composants appropriés, activer le REST API, configurer la sécurité, et démarrer le serveur mqweb.

Avant de commencer

IBM i Sous IBM i, les commandes doivent être exécutées dans QSHHELL.

Pourquoi et quand exécuter cette tâche

La procédure de cette tâche se concentre sur l'initiation rapide à administrative REST API. Les étapes de configuration de la sécurité expliquent comment configurer un registre d'utilisateurs de base, mais il existe d'autres options de configuration des utilisateurs et des rôles. Pour plus d'informations sur la configuration de la sécurité pour le administrative REST API, voir [IBM MQ Console and REST API security](#).

Remarque : Vous devez être un [utilisateur privilégié](#) pour accéder au fichier `mqwebuser.xml` .

Procédure

1. Installez le composant IBM MQ Console et REST API :

- **V 9.0.4** **AIX** Sous AIX, installez l'ensemble de fichiers `mqm.web.rte` .
- **Linux** Sous Linux, installez le composant `MQSeriesWeb` . Pour plus d'informations sur l'installation de composants et de fonctions sous Linux, voir [Tâches d'installation d'Linux](#).
- **Windows** Sous Windows, installez la fonction `Web Administration` . Pour plus d'informations sur l'installation de composants et de fonctions sous Windows, voir [Tâches d'installation d'Windows](#).

- **z/OS** Sous z/OS, installez la fonction IBM MQ for z/OS Unix System Services Web Components . Pour plus d'informations sur l'installation de composants et de fonctions sous z/OS, voir [Tâches d'installation d'z/OS](#).
2. Facultatif : Depuis la IBM MQ 9.0.2, l'administrative REST API est activée par défaut. Toutefois, dans IBM MQ 9.0.1, administrative REST API est désactivé par défaut. Si vous utilisez IBM MQ 9.0.1, activez REST API en ajoutant le code XML suivant au fichier `mqwebuser.xml` , dans les balises `<server>` :

```
<variable name="mqRestAutostart" value="true"/>
```

Le fichier `mqwebuser.xml` se trouve dans l'un des répertoires suivants:

- **ULW** `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **z/OS** `WLP_user_directory/servers/mqweb`

où `WLP_user_directory` est le répertoire qui a été spécifié lors de l'exécution du script `crtmqweb.sh` pour créer la définition de serveur `mqweb`.

3. **V 9.0.2**

Depuis la IBM MQ 9.0.2, elle est activée pour l'administrative REST API. Vous devez configurer les utilisateurs et les rôles pour pouvoir utiliser administrative REST API:

- a) Copiez le fichier `basic_registry.xml` depuis le répertoire `MQ_INSTALLATION_PATH/web/mq/samp/configuration` .
- b) Placez l'exemple de fichier XML dans le répertoire approprié:

- **ULW** Sous UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **z/OS** Sous z/OS : `WLP_user_directory/servers/mqweb`

où `WLP_user_directory` est le répertoire qui a été spécifié lors de l'exécution du script `crtmqweb.sh` pour créer la définition du serveur `mqweb`.

- c) Renommez l'exemple de fichier XML en `mqwebuser.xml`.

Remarque : Ce fichier renommé remplace un fichier existant qui est également utilisé pour IBM MQ Console. Par conséquent, si vous avez modifié le fichier `mqwebuser.xml` pour IBM MQ Console, copiez vos modifications dans le nouveau fichier XML avant de le renommer.

- d) Facultatif : Editez le fichier `mqwebuser.xml` pour ajouter des utilisateurs et des groupes. Affectez à ces utilisateurs et à ces groupes les rôles appropriés pour être autorisés à utiliser le administrative REST API. Vous pouvez également modifier les mots de passe des utilisateurs définis par défaut et coder les nouveaux mots de passe. Pour plus d'informations, voir [Configuration des utilisateurs et des rôles](#).

4. Activez les connexions distantes au serveur `mqweb`:

- A partir de la version 9.0.4, utilisez la commande **setmqweb** :
- ```
setmqweb properties -k httpHost -v hostname
```
- A partir de la version 9.0.1, ajoutez le code XML suivant au fichier `mqwebuser.xml` , dans les balises `<server>` :

```
<variable name="httpHost" value="hostname"/>
```

où `nom_hôte` indique l'adresse IP, le nom d'hôte du serveur de noms de domaine (DNS) avec le suffixe du nom de domaine ou le nom d'hôte DNS du serveur sur lequel IBM MQ est installé. Utilisez un astérisque (\*) pour indiquer toutes les interfaces réseau disponibles.



**Avertissement :**

**z/OS**

**V 9.0.4**

Avant d'émettre la commande **setmqweb** ou **dspmqweb** sous z/OS, vous devez définir la variable d'environnement WLP\_USER\_DIR de sorte qu'elle pointe vers votre configuration de serveur mqweb.

Pour ce faire, entrez la commande suivante :


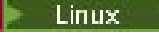

```
export WLP_USER_DIR=WLP_user_directory
```

où *WLP\_user\_directory* est le nom du répertoire transmis à `crtmqweb.sh`. Exemple :





```
export WLP_USER_DIR=/var/mqm/web/installation1
```

Pour plus d'informations, voir [Création de la définition de serveur Liberty](#).

5. Démarrez le serveur mqweb qui prend en charge le REST API:

-   Sous Windows et Linux, en tant qu' utilisateur privilégié, entrez la commande suivante sur la ligne de commande:  
`strmqweb`
-  Sous z/OS, démarrez la procédure que vous avez créée dans [Tâche 29: Créer une procédure pour le IBM WLP](#).

## Que faire ensuite

1.  Choisissez comment les utilisateurs de administrative REST API s'authentifient auprès du serveur mqweb. Il n'est pas nécessaire d'utiliser la même méthode pour tous les utilisateurs. Les options suivantes sont disponibles :
  - Permet aux utilisateurs de s'authentifier à l'aide de l'authentification de base HTTP. Dans ce cas, un nom d'utilisateur et un mot de passe sont codés, mais non chiffrés, et envoyés avec chaque demande REST API pour authentifier et autoriser l'utilisateur pour cette demande. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir [Using HTTP basic authentication with the REST API](#).
  - Permet aux utilisateurs de s'authentifier à l'aide de l'authentification par jeton. Dans ce cas, un utilisateur fournit un ID utilisateur et un mot de passe à la ressource REST API `login` avec la méthode HTTP POST. Un jeton LTPA est généré pour permettre à l'utilisateur de rester connecté et autorisé pendant une durée définie. Pour que cette authentification soit sécurisée, vous devez utiliser une connexion sécurisée. C'est-à-dire que vous devez utiliser HTTPS. Pour plus d'informations, voir [Utilisation de l'authentification basée sur un jeton avec REST API](#).
  - Permet aux utilisateurs de s'authentifier à l'aide de certificats client. Dans ce cas, l'utilisateur n'utilise pas d'ID utilisateur ou de mot de passe pour se connecter à administrative REST API, mais utilise le certificat client à la place. Pour plus d'informations, voir [Utilisation de l'authentification par certificat client avec REST API](#).
2.  Configurez les paramètres REST API , y compris l'activation des connexions HTTP et la modification du numéro de port. Pour plus d'informations, voir [Configuration d' IBM MQ Console et REST API](#).
3.  Vous pouvez éventuellement configurer le REST API pour MFT. Pour plus d'informations, voir la section «[Configuration requise pour MFT REST API](#)», à la page 76.
4.  Vous pouvez éventuellement configurer le partage de ressources d'origine croisée pour REST API. Par défaut, vous ne pouvez pas accéder au REST API à partir de ressources Web qui ne sont pas hébergées sur le même domaine que le REST API. C'est-à-dire que les demandes d'origine croisée ne sont pas activées. Vous pouvez configurer le partage de ressources d'origine croisée (CORS) pour autoriser les demandes d'origine croisée à partir d'URL spécifiées. Pour plus d'informations, voir [Configuration de CORS pour REST API](#).

5. Utilisez le REST API. Pour plus d'informations, voir «Utilisation de administrative REST API», à la page 79, «Administration à distance à l'aide de REST API», à la page 85, et la référence d'administration REST API.

**Remarque :** Vous pouvez arrêter le serveur mqweb à tout moment à l'aide de la commande **endmqweb** . Toutefois, si le serveur mqweb n'est pas en cours d'exécution, vous ne pouvez pas utiliser REST API ou IBM MQ Console.

## **V 9.0.5** Configuration requise pour MFT REST API

Procédure à suivre pour configurer Managed File Transfer afin d'utiliser REST API

Dans le fichier `mqwebuser.xml` , définissez les éléments suivants:

- La propriété **mqRestMftEnabled** doit avoir la valeur *true*

**Remarque :** Vous devez redémarrer le serveur mqweb si vous modifiez la valeur de cette propriété.

- **mqRestMftCoordinationQmgr** à un gestionnaire de files d'attente de coordination MFT approprié s'exécutant localement sur la machine sur laquelle le serveur mqweb s'exécute.

Pour qu'un gestionnaire de files d'attente agisse en tant que gestionnaire de files d'attente de coordination MFT pour REST API, vous devez exécuter la commande suivante:

- Commande **fteSetupCoordination** sur le même gestionnaire de files d'attente local que celui défini pour la propriété **mqRestMftCoordinationQmgr** dans le fichier `mqwebuser.xml` .

Cette commande génère un fichier MQSC qui contient les définitions des objets IBM MQ . Notez que la commande peut être exécutée sur n'importe quelle machine sur laquelle MFT est installé.

- Le fichier MQSC généré à l'étape précédente doit être exécuté sur le gestionnaire de files d'attente de coordination pour créer les objets IBM MQ requis.

### Information associée

[Configuration de la sécurité MFTREST API](#)

[Administration de MFT REST API](#)

[GET-liste des transferts](#)

[GET-statut de transfert](#)

## **V 9.0.1** Détermination de l'URL REST API

L'URL par défaut permettant d'accéder à administrative REST API à partir de IBM MQ 9.0.4 est: `https://localhost:9443/ibmmq/rest/v1/admin`. Dans IBM MQ 9.0.3 et les versions antérieures, l'URL par défaut est: `https://localhost:9443/ibmmq/rest/v1`. L'URL par défaut permettant d'accéder à messaging REST API est: `https://localhost:9443/ibmmq/rest/v1/messaging`. Si l'hôte ou le port par défaut est modifié ou si HTTP est activé, vous pouvez déterminer l'URL à l'aide de la commande **dspmqweb** .




### Pourquoi et quand exécuter cette tâche

**V 9.0.4** Depuis la IBM MQ 9.0.4, vous pouvez utiliser la commande **dspmqweb status** pour déterminer l'URL REST API sous Windows, Linux et z/OS.

Pour IBM MQ 9.0.3 et les versions antérieures, sous Windows et Linux, vous pouvez utiliser la commande **dspmqweb** . Sous z/OS , vous devez effectuer une recherche dans le fichier `messages.log` pour déterminer l'URL.

### Procédure

- **V 9.0.4**  
Utilisez l'une des méthodes suivantes pour déterminer l'URL:

- Depuis IBM MQ 9.0.4, utilisez la commande **dspmweb status** en tant qu' utilisateur privilégié:
  1. Vérifiez que le serveur mqweb est en cours d'exécution en entrant la commande **strmqweb** sur la ligne de commande.
  2. Affichez l'URL en entrant la commande `dspmweb status` sur la ligne de commande.
- Pour IBM MQ 9.0.3 et les versions antérieures, utilisez l'une des méthodes suivantes:
  -   Sous Windows ou Linux, utilisez la commande **dspmweb** en tant qu' utilisateur privilégié:
    1. Vérifiez que le serveur mqweb est en cours d'exécution en entrant la commande **strmqweb** sur la ligne de commande.
    2. Affichez l'URL en entrant la commande `dspmweb` sur la ligne de commande.
  -  Sous z/OS, utilisez le fichier `messages.log` pour rechercher l'URL:
    1. Ouvrez le fichier `messages.log`.  
Le fichier `messages.log` se trouve dans le chemin `WLP_user_directory/servers/mqweb/logs`, où `WLP_user_directory` est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb.sh** pour créer la définition de serveur mqweb.
    2. Recherchez le dernier code de message CWWKT0016I qui se termine par `ibmmq/rest/v1`. L'URL est incluse dans ce message.

## Exemple

L'exemple suivant illustre le code de message CWWKT0016I avec l'URL comme URL par défaut:

```
A CWWKT0016I: Web application available (default_host): https://localhost:9443/ibmmq/rest/v1
```

## Vérification de la configuration correcte du serveur mqweb sous z/OS

Étapes à effectuer pour vérifier que le serveur mqweb est correctement configuré sous z/OS et pour résoudre les problèmes de configuration courants.

## Procédure

1. Vérifiez la tâche démarrée du serveur mqweb .  
Vous devriez voir des messages de journal de travail similaires aux suivants:

```
+CWWKE0001I: The server mqweb has been launched.
+CWWKZ0001I: Application com.ibm.mq.rest started in 1.493 seconds.
+CWWKZ0001I: Application com.ibm.mq.console started in 0.885 seconds.
+CWWKF0011I: The server mqweb is ready to run a smarter planet.
```

Il ne doit pas y avoir de messages dans STDERR

Il doit y avoir des messages dans STDOUT similaires aux suivants:

```
Launching mqweb (WebSphere Application Server 17.0.0.2, WAS FOR Z/OS 17.0.0.2,
MQM MVS/ESA V9 R0.4/wlp-1.0.17.c1170220170523-1818) on IBM J9 VM,
version pmz6480sr4fp7-20170627_02 (SR4 FP7) (en_US)
[AUDIT] CWWKE0001I: The server mqweb has been
launched.
[AUDIT] CWWKG0028A: Processing included configuration resource: /mqm/V9R0M4/web/mq/etc/
mqweb.xml
[AUDIT] CWWKG0028A: Processing included configuration resource:
var/mqm/mqweb904/servers/mqweb/mqwebuser.xml
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/api/docs/
[AUDIT] CWWKT0016I: Web application available (default_host):
```

```

yourhost:yourport/api/explorer/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibm/api/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibm/api/explorer/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibm/api/docs/subscription/websocket/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport:9080/ibmmq/rest/v1/
[AUDIT] CWWKZ0001I: Application com.ibm.mq.rest started in 1.493 seconds.
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibmmq/console/
[AUDIT] CWWKT0016I: Web application available (default_host):
yourhost:yourport/ibmmq/console/internal/
[AUDIT] CWWKZ0001I: Application com.ibm.mq.console started in 1.459 seconds.
[AUDIT] CWWKF0012I: The server installed the following features:
[jsr-2.2, servlet-3.1, ssl-1.0, jndi-1.0, basicAuthenticationMQ-1.0,
apiDiscovery-1.0, localConnector-1.0, appSecurity-2.0, jaxrs-1.1,
concurrent-1.0, json-1.0, websocket-1.0, distributedMap-1.0,
applicationMonitorMQ-1.0].
[AUDIT] CWWKF0011I: The server mqweb is ready to run a smarter planet.
[AUDIT] MQWB2019I: MQ Console level: 9.0.4 - V904-GA904-L171016
[AUDIT] MQWB0023I: MQ REST API level: 9.0.4 - V904-GA904-L171016

```

### Remarques :

- a. Si la procédure ne parvient pas à démarrer, résolvez les problèmes.
- b. Si les messages sont similaires à:

```

Web application available (default_host):
http://localhost:portnumber/ibmmq/console/

```

vous n'avez pas défini la propriété **httpHost** pour autoriser les connexions distantes au serveur mqweb .

```
<variable name="httpHost" value="*" />
```

- c. Si vous avez modifié la propriété **httpHost** , un message similaire au suivant s'affiche:

```

Web application available (default_host):
yourhost:portnumber/ibmmq/console/

```

En fonction de votre configuration TCP/IP, il se peut que vous ne puissiez pas utiliser l'adresse affichée en tant qu'URL.

- d. Les modifications apportées aux fichiers de configuration du serveur XML sont prises en compte quelques secondes plus tard. Il n'est pas nécessaire de redémarrer le serveur mqweb
2. Connectez-vous à IBM MQ Console.

Vous devriez voir une fenêtre s'afficher avec le titre IBM MQ Console.

### Remarques :

- a. Si vous voyez une fenêtre avec `This site can't be reached` ou `Context Root Not Found`, IBM MQ Console n'est pas actif ; vous devez patienter.
- b. Si aucune fenêtre ne s'affiche avec le titre IBM MQ Console, il se peut que des informations de diagnostic supplémentaires soient disponibles dans `WLP_USER_DIRECTORY/servers/mqweb/logs/messages.log`, où `WLP_USER_DIRECTORY` correspond au répertoire que vous avez spécifié, lorsque vous avez exécuté le script **crtmqweb.sh** pour créer la définition de serveur mqweb .

Notez que ce fichier est en ASCII et pour l'afficher, vous pouvez utiliser **oddit** à partir d'une ligne de commande USS ou utiliser l'option ISPF 3.17 et la commande de ligne **va** (afficher ASCII).

- c. Vous pouvez obtenir des messages dans STDOUT similaires à:

```

[WARNING] SRVE0190E: File not found: /nls/en_GB/labels.json
[WARNING] SRVE0190E: File not found: /nls/en_GB/errors.json
[WARNING] SRVE0190E: File not found: /nls/en_GB/strings.json

```

```
[WARNING] SRVE0190E: File not found: /nls/en_GB/pcf.json
```

Vous pouvez ignorer ces messages.

- d. Si des gestionnaires de files d'attente sont définis sur le système sur lequel le serveur mqweb est en cours d'exécution et qu'ils s'exécutent au même niveau que le serveur mqweb, ils s'affichent dans le widget **Gestionnaires de files d'attente locaux** de MQ Console.

Si aucun gestionnaire de files d'attente n'est affiché, vous ne disposez d'aucun gestionnaire de files d'attente au même niveau que le serveur mqweb, qui a été démarré depuis le dernier démarrage du système.

3. Si vous recevez un message:

Lost communication with the server Could not establish communication with the server.

- a) Dans la procédure utilisée pour démarrer le serveur mqweb :

- i) Vérifiez que les bibliothèques STEPLIB sont au niveau correct et qu'elles sont autorisées par APF.
- ii) Vérifiez que les points PATH et LIBPATH pointent vers le chemin correct. Les valeurs PATH et LIBPATH sont définies dans la procédure de tâche démarrée de votre serveur Web, que vous utilisez sur la base de l'exemple CSQ4WEBS.

- b) Dans USS, utilisez la commande `ls -ltr PathPrefix/web/bin/dspmq`, où PathPrefix est le chemin d'installation de IBM MQ Unix System Services Components.

Ceci doit afficher une sortie similaire à `-rwxr-xr-t a-s- ... /mqm/V9R0M4/web/bin/dspmq`

Vérifiez que les indicateurs t et a sont définis.

Entrez la commande :

- `chmod +t PathPrefix/web/bin/dspmq` pour définir le bit de rappel (t)
- `extattr +a PathPrefix/web/bin/dspmq` pour définir l'attribut avec droits APF (a)

## V 9.0.1 Utilisation de administrative REST API

Lorsque vous utilisez administrative REST API, vous appelez des méthodes HTTP sur des URL qui représentent les différents objets IBM MQ, tels que les gestionnaires de files d'attente ou les files d'attente. La méthode HTTP, par exemple POST, représente le type d'action à effectuer sur l'objet représenté par l'URL. Des informations supplémentaires sur l'action peuvent être fournies en JSON dans le cadre du contenu de la méthode HTTP ou codées dans des paramètres de requête. Des informations sur le résultat de l'exécution de l'action peuvent être renvoyées en tant que corps de la réponse HTTP.

### Avant de commencer

Tenez compte des points suivants avant d'utiliser administrative REST API:

- **V 9.0.2** Vous devez vous authentifier auprès du serveur mqweb pour pouvoir utiliser administrative REST API. Vous pouvez vous authentifier à l'aide de l'authentification de base HTTP, de l'authentification par certificat client ou de l'authentification basée sur un jeton. Pour plus d'informations sur l'utilisation de ces méthodes d'authentification, voir [IBM MQ Console et REST API security](#).
- REST API est sensible à la casse. Par exemple, une requête HTTP GET sur l'URL suivante n'affiche pas d'informations si le gestionnaire de files d'attente est appelé qmgr1.

```
/ibmmq/rest/v1/admin/qmgr/QMGR1
```

- Tous les caractères pouvant être utilisés dans les noms d'objet IBM MQ ne peuvent pas être directement codés dans une URL. Pour coder correctement ces caractères, vous devez utiliser le codage d'URL approprié:
  - Une barre oblique (/) doit être codée sous la forme %2F.

- Le signe de pourcentage,%, doit être codé en tant que %25.
- En raison du comportement de certains navigateurs, ne nommez pas les objets en utilisant uniquement des points ou des barres obliques.

## Pourquoi et quand exécuter cette tâche

Lorsque vous utilisez REST API pour effectuer une action sur un objet, vous devez d'abord construire une URL pour représenter cet objet. Chaque URL commence par un préfixe, qui décrit le nom d'hôte et le port auxquels envoyer la demande. Le reste de l'URL décrit un objet particulier, ou un ensemble d'objets, appelé ressource.

L'action à effectuer sur la ressource définit si l'URL a besoin ou non de paramètres de requête. Il définit également la méthode HTTP utilisée et indique si des informations supplémentaires sont envoyées à l'URL, ou renvoyées à partir de celle-ci, au format JSON. Les informations supplémentaires peuvent faire partie de la demande HTTP ou être renvoyées dans le cadre de la réponse HTTP.

Après avoir construit l'URL et créé un contenu JSON facultatif pour l'envoi dans la demande HTTP, vous pouvez envoyer la demande HTTP à IBM MQ. Vous pouvez envoyer la demande à l'aide de l'implémentation HTTP qui est intégrée au langage de programmation de votre choix. Vous pouvez également envoyer les demandes à l'aide d'outils de ligne de commande tels que cURL, d'un navigateur Web ou d'un module complémentaire de navigateur Web.

**Important :** Vous devez au minimum effectuer les étapes [«1.a», à la page 80](#) et [«1.b», à la page 80](#).

## Procédure

1. Construisez l'URL:

a) Commencez par l'URL de préfixe suivante:

**V 9.0.4** IBM MQ 9.0.4 et versions ultérieures :

```
https://host:port/ibmmq/rest/v1/admin
```

IBM MQ 9.0.3 et versions antérieures :

```
https://host:port/ibmmq/rest/v1
```

### host

Indique le nom d'hôte ou l'adresse IP sur lequel administrative REST API est disponible.

La valeur par défaut est localhost.

### port

Indique le numéro de port HTTPS utilisé par administrative REST API .

La valeur par défaut est 9443.

**V 9.0.1** Si vous activez les connexions HTTP, vous pouvez utiliser HTTP au lieu de HTTPS. Pour plus d'informations sur l'activation de HTTP, voir [Configuration des ports HTTP et HTTPS](#).

Pour plus d'informations sur la détermination de l'URL de préfixe, voir [«Détermination de l'URL REST API», à la page 76](#).

b) Ajoutez la ressource au chemin d'URL.

Les ressources suivantes sont disponibles:

- [installation](#)
- [qmgr](#)
- **V 9.0.2** [File d'attente](#)
- **V 9.0.4** [abonnement](#)



#### ▶ V 9.0.4 Canal

Par exemple, pour interagir avec les gestionnaires de files d'attente, ajoutez /qmgr à l'URL de préfixe pour créer l'URL suivante:

#### ▶ V 9.0.4 IBM MQ 9.0.4 et versions ultérieures :

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr
```

IBM MQ 9.0.3 et versions antérieures :

```
https://localhost:9443/ibmmq/rest/v1/qmgr
```

c) Facultatif : Ajoutez des segments de chemin facultatifs supplémentaires à l'URL.

Dans les informations de référence de chaque type d'objet, les segments facultatifs peuvent être identifiés dans l'URL par les accolades qui l'entourent { }.

Par exemple, ajoutez le nom de gestionnaire de files d'attente QM1 à l'URL pour créer l'URL suivante:

#### ▶ V 9.0.4 IBM MQ 9.0.4 et versions ultérieures :

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1
```

IBM MQ 9.0.3 et versions antérieures :

```
https://localhost:9443/ibmmq/rest/v1/qmgr/QM1
```

d) Facultatif : Ajoutez un paramètre de requête facultatif à l'URL.

Ajouter un point d'interrogation,?, nom de variable, signe égal =, et une valeur ou une liste de valeurs à l'URL.

Par exemple, pour demander tous les attributs du gestionnaire de files d'attente QM1, créez l'URL suivante:

#### ▶ V 9.0.4 IBM MQ 9.0.4 et versions ultérieures :

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM1?attributes=*
```

IBM MQ 9.0.3 et versions antérieures :

```
https://localhost:9443/ibmmq/rest/v1/qmgr/QM1?attributes=*
```

e) Ajoutez d'autres paramètres de requête facultatifs à l'URL.

Ajoutez une perluète, &, à l'URL, puis répétez l' étape d.

2. Appelez la méthode HTTP appropriée sur l'URL. Spécifiez le contenu JSON facultatif et fournissez les données d'identification de sécurité appropriées pour authentifier. Exemple :

- Utilisez l'implémentation HTTP/REST du langage de programmation de votre choix.
- Utilisez un outil tel qu'un module complémentaire de navigateur client REST ou cURL.

## ▶ V 9.0.5 Obtention d'une liste MFT de transferts ou d'un statut de transfert

Obtention de la liste de transferts Managed File Transfer à l'aide de la REST API

Pour plus d'informations, voir [Administration à l'aide de REST API](#).

Pour obtenir la liste des transferts, procédez comme suit:

1. Envoyez une demande [GET](#) au serveur qui héberge l'application Web MFT etMFT . Vous devez prendre en compte le format de l'URL lors de l'envoi d'une demande.

Pour plus d'informations, voir [admin / mft/transfer](#) .

2. Lorsqu'une demande est reçue, une vérification est effectuée sur la demande pour déterminer si elle est valide.

Pour plus d'informations, voir [Configuration de la sécurité MFT REST API](#) .

3. Si la demande est valide, vous recevez une réponse positive avec le corps de la réponse. Sinon, un code d'erreur et un code de réponse appropriés sont générés.

Pour obtenir la liste des réponses, voir [Codes de statut de la réponse](#) .

Pour obtenir le statut d'un transfert, procédez comme suit:

1. Envoyez une demande `GET` au serveur qui héberge l'application Web MFT etMFT . Vous devez prendre en compte le format de l'URL lors de l'envoi d'une demande.

Pour plus d'informations, voir [admin / mft/transfer](#) .

2. Lorsqu'une demande est reçue, une vérification est effectuée sur la demande pour déterminer si elle est valide.

Pour plus d'informations, voir [Configuration de la sécurité MFT REST API](#) .

3. Si la demande est valide, vous recevez une réponse positive avec le corps de la réponse. Sinon, un code d'erreur et un code de réponse appropriés sont générés.

Pour obtenir la liste des réponses, voir [Codes de statut de la réponse](#) .

## Exemple de corps de réponse

La liste de transfert est extraite du stockage interne et un corps de réponse est généré au format JSON pour chacun des transferts. Le corps de réponse d'un transfert se présente comme suit:

```
{
 "transfer": [
 {
 "destinationAgent": {
 "name": "AGENT.TRI.BANK"
 },
 "originator": {
 "host": "192.168.99.1",
 "userId": "johndoe"
 },
 "sourceAgent": {
 "name": "TESTAGENT"
 },
 "statistics": {
 "endTime": "2018-01-08T16:22:15.569Z",
 "numberOfFileFailures": 0,
 "numberOfFileSuccesses": 2,
 "numberOfFileWarnings": 0,
 "numberOfFiles": 2,
 "startTime": "2018-01-08T16:22:15.242Z"
 },
 "status": {
 "state": "successful"
 },
 "id": "414D51204D465444454D4F3320202020513E525A21109908"
 },
 {
 "destinationAgent": {
 "name": "AGENT.TRI.BANK"
 },
 "originator": {
 "host": "192.168.99.1",
 "userId": "ramsubbarao"
 },
 "sourceAgent": {
 "name": "TESTAGENT"
 },
 "statistics": {
 "endTime": "2018-01-08T16:22:13.573Z",
 "numberOfFileFailures": 0,
 "numberOfFileSuccesses": 2,
 "numberOfFileWarnings": 0,
 "numberOfFiles": 2,
 "startTime": "2018-01-08T16:22:13.167Z"
 },
 "status": {
 "state": "successful"
 },
 "id": "414D51204D465444454D4F3320202020513E525A21109702"
 }
]
}
```

```
}
}]}
```

Voir [Attributs de corps de réponse JSON REST API](#) pour plus d'informations sur les attributs.

## V 9.0.5 Obtention du statut de l'agent MFT

Comment obtenir le statut de l'agent Managed File Transfer à l'aide de la REST API

Pour plus d'informations, voir [Administration à l'aide de REST API](#).

Pour obtenir le statut de l'agent, procédez comme suit:

1. Envoyez une demande [GET](#) à l'agent sous le gestionnaire de files d'attente de coordination par défaut qui utilise l'application Web MFT etMFT . Vous devez prendre en compte le format de l'URL lors de l'envoi d'une demande.

Pour plus d'informations, voir [admin / mft/agent](#) .

2. Lorsqu'une demande est reçue, une vérification est effectuée sur la demande pour déterminer si elle est valide.

Pour plus d'informations, voir [Configuration de la sécurité MFT REST API](#) .

3. Si la demande est valide, vous recevez une réponse positive avec le corps de la réponse. Sinon, un code d'erreur et un code de réponse appropriés sont générés.

Pour obtenir la liste des réponses, voir [Codes de statut de la réponse](#) .

## Exemple d'URL

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent**

Génère une liste des **name**, **state** et **type** de tous les agents sous le gestionnaire de files d'attente de coordination.

Pour plus d'informations sur ces attributs, voir [Attributs de statut de l'agent REST API](#) .

Si vous ajoutez `?attributes=*` à la fin de la chaîne d'URL, la sortie contient une liste de tous les attributs de chaque agent qui se trouve sous le gestionnaire de files d'attente de coordination.

Si vous ajoutez `/ {agentName}` à la fin de la chaîne d'URL, la sortie contient les éléments **name**, **state** et **type** de l'agent qui correspond à `agentName`.

De même, si vous ajoutez `/ {agentName}?attributes=*` à la fin de la chaîne d'URL, la sortie contient une liste de tous les attributs de l'agent qui correspondent à `agentName`.

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent?name={<prefix>\*<suffix> }**

Par exemple, `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?name=AGENT*TEST` génère le récapitulatif d'attribut par défaut pour les agents commençant par le nom `AGENT` et se terminant par le nom `TEST`.

Si vous omettez `*TEST` dans l'URL, la sortie contient le récapitulatif d'attribut par défaut pour tous les agents commençant par le nom `AGENT`.

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type={agentType}**

Par exemple, `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type=standard` génère le récapitulatif d'attribut par défaut pour les agents de type `standard`.

De même, si vous remplacez `type={agentType}` par `state={agentState}`, la sortie affiche le récapitulatif des attributs par défaut pour les agents de l'état spécifié.

### **{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type={agentType}&attributes=\***

Par exemple, `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type=standard&attributes=*` génère tous les attributs d'un agent dont le type est `standard`.

Vous pouvez également effectuer un filtrage en fonction d'une combinaison d'attributs. Exemple :

- `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?name=*bob&type=standard&state=ready&attributes=*`

- `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?type=standard&state=ready&attributes=*`
- `{baseURI}/ibmmq/rest/{version}/admin/mft/agent?name=agent*&state=ready&attributes=*`

## Exemple de corps de réponse

Pour plus d'informations sur les attributs décrits, voir [Attributs de statut de l'agent REST API](#) et [Attributs de corps de réponse du statut de l'agent REST API](#).

L'exécution de la commande suivante, `/ibmmq/rest/v1/admin/mft/agent/`, répertorie le statut de l'agent de base du nom, du type et de l'état de l'agent. Exemple :

```
{
 "agent": [
 { "name": "AGENT1",
 "state": "ready",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "ready",
 "type": "standard" },
 { "name": "BRIDGE_AGENT3",
 "type": "protocolBridge",
 "state": "ready" },
 { "name": "CD_AGENT",
 "type": "connectDirectBridge",
 "state": "ready" }]
}
```

Emission de chacune des commandes suivantes:

```
/ibmmq/rest/v1/admin/mft/agent?type=standard
/ibmmq/rest/v1/admin/mft/agent?state=stopped
/ibmmq/rest/v1/admin/mft/agent?name=AGENT*
```

répertorie les informations de base d'un agent correspondant au nom, à l'état et au type particuliers de chaque commande. Exemple :

```
{ "agent": [
 { "name": "AGENT1",
 "state": "ready",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "ready",
 "type": "standard" }]
}
{ "agent": [
 { "name": "AGENT1",
 "state": "stopped",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "stopped",
 "type": "standard" }]
}
{ "agent": [
 { "name": "AGENT1",
 "state": "ready",
 "type": "standard" },
 { "name": "AGENT2",
 "state": "ready",
 "type": "standard" }]
}
```

En exécutant la commande suivante, `/ibmmq/rest/v1/admin/mft/agent?attributes=general&type=standard` répertorie tous les messages de type *standard*, ainsi que les attributs *general*. Exemple :

```
{
 "agent": [
 { "name": "AGENT1",
 "state": "ready",
 "type": "standard",
 "general": { "description": "Standard connected to the qmgr in client mode",
```

```

 "statusAge" : "06:31:00",
 "version" : "9.0.3.0",
 "level" : "p903-L170513",
 "statusPublicationRate" : 300,
 "statusPublishTime" : "2017-10-31T06:57:07.000Z",
 "maximumQueuedTransfers" : 1000,
 "maximumDestinationTransfers":25,
 "maximumSourceTransfers":25,
 "operatingSystem" : "Windows7" }
 },
 { "name": "AGENT2",
 "state" : "ready",
 "type": "standard"
 "general" : { "description" : "Standard connected to qmgr in Binding mode",
 "statusAge" : "05:00:00",
 "version" : "9.0.3.0",
 "level" : "p903-L170513",
 "statusPublicationRate" : 300,
 "statusPublishTime" : "2017-09-13T09:10:09.000Z",
 "maximumQueuedTransfers" : 1000,
 "maximumDestinationTransfers":25,
 "maximumSourceTransfers":25,
 "operatingSystem" : "Windows7" }
 }
]
}

```

## V 9.0.4 Administration à distance à l'aide de REST API



Vous pouvez utiliser REST API pour administrer les gestionnaires de files d'attente éloignées et les objets IBM MQ qui sont associés à ces gestionnaires de files d'attente. Cette administration à distance inclut les gestionnaires de files d'attente qui se trouvent sur le même système, mais pas dans la même installation IBM MQ que le serveur mqweb. Par conséquent, vous pouvez utiliser l'REST API pour administrer l'intégralité de votre réseau IBM MQ avec une seule installation qui exécute le serveur mqweb. Pour administrer les gestionnaires de files d'attente éloignées, vous devez configurer la passerelle administrative REST API de sorte qu'au moins un gestionnaire de files d'attente de la même installation que le serveur mqweb fasse office de gestionnaire de files d'attente de passerelle. Vous pouvez ensuite spécifier le gestionnaire de files d'attente éloignées dans l'URL de ressource REST API pour effectuer l'action d'administration spécifiée.

### Avant de commencer

Vous pouvez empêcher l'administration à distance en désactivant la passerelle administrative REST API . Pour plus d'informations, voir [Configuration de la passerelle administrative REST API](#).

Pour utiliser la passerelle administrative REST API , les conditions suivantes doivent être remplies:

- Le serveur mqweb doit être configuré et démarré. Pour plus d'informations sur la configuration et le démarrage du serveur mqweb, voir [«Guide d'initiation à administrative REST API»](#), à la page 73.
- Le gestionnaire de files d'attente que vous souhaitez configurer en tant que gestionnaire de files d'attente de passerelle doit se trouver dans la même installation que le serveur mqweb.
- Le gestionnaire de files d'attente éloignées que vous souhaitez administrer doit être au niveau IBM MQ 8.0 ou ultérieur.
- Vous devez vous assurer que tous les attributs spécifiés dans votre demande sont valides pour le système auquel vous envoyez la demande. Par exemple, si le gestionnaire de files d'attente de passerelle est sous Windows et que le gestionnaire de files d'attente éloignées est sous z/OS, vous ne pouvez pas demander que l'attribut `dataCollection.statistics` soit renvoyé pour une demande HTTP GET sur la ressource `queue` .
- Vous devez vous assurer que tous les attributs spécifiés dans votre demande sont valides pour le niveau de IBM MQ auquel vous envoyez la demande. Par exemple, si le gestionnaire de files d'attente éloignées exécute IBM MQ 8.0, vous ne pouvez pas demander que l'attribut `extended.enableMediaImageOperations` soit renvoyé pour une demande HTTP GET sur la ressource `queue` .
- Vous devez utiliser l'une des ressources REST prises en charge suivantes:

- /queue
- /subscription
-  /channel
-  /mqsc
- /qmgr

La ressource /qmgr renvoie uniquement un sous-ensemble des attributs lorsque vous interrogez un gestionnaire de files d'attente éloignées: name, status.started, status.channelInitiatorState, status.ldapConnectionState, status.connectionCount et status.publishSubscribeState.

## Pourquoi et quand exécuter cette tâche

Pour utiliser la passerelle administrative REST API afin d'administrer les gestionnaires de files d'attente éloignées, vous devez préparer les gestionnaires de files d'attente pour l'administration à distance. Autrement dit, vous devez configurer les files d'attente de transmission, les programmes d'écoute et les canaux émetteur et récepteur entre le gestionnaire de files d'attente de passerelle et le gestionnaire de files d'attente éloignées. Vous pouvez ensuite envoyer une demande REST au gestionnaire de files d'attente éloignées en spécifiant le gestionnaire de files d'attente dans l'URL de la ressource. Le gestionnaire de files d'attente de passerelle est spécifié en définissant l'attribut mqRestGatewayQmgr à l'aide de la commande **setmqweb** sur le nom du gestionnaire de files d'attente de passerelle ou en envoyant le nom du gestionnaire de files d'attente de passerelle dans un en-tête envoyé avec la demande. La demande est envoyée via le gestionnaire de files d'attente de passerelle au gestionnaire de files d'attente éloignées. La réponse est renvoyée avec un en-tête indiquant le gestionnaire de files d'attente qui a été utilisé comme gestionnaire de files d'attente de passerelle.

## Procédure

1. Configurez les communications entre le gestionnaire de files d'attente de passerelle et les gestionnaires de files d'attente éloignées que vous souhaitez administrer. Ces étapes de configuration sont les mêmes que celles requises pour configurer l'administration à distance par runmqsc et PCF.  
Pour plus d'informations sur ces étapes, voir [«Administration à distance à partir d'un gestionnaire de files d'attente local»](#), à la page 201.
2. Configurez la sécurité sur les gestionnaires de files d'attente éloignées:
  - a) Vérifiez que les ID utilisateur appropriés existent sur le système sur lequel s'exécute le gestionnaire de files d'attente éloignées. L'ID utilisateur qui doit exister sur le système distant dépend du rôle de l'utilisateur REST API :
    - Si l'utilisateur REST API se trouve dans le groupe MQWebAdmin ou MQWebAdminRO, l'ID utilisateur qui a démarré le serveur mqweb doit exister sur le système distant. Sur le IBM MQ Appliance, l'utilisateur qui démarre le serveur mqweb est mqsystem.
    - Si l'utilisateur REST API appartient au groupe MQWebUser, cet ID utilisateur REST API doit exister sur le système distant.
  - b) Assurez-vous que les ID utilisateur appropriés disposent des niveaux d'autorisation nécessaires pour accéder aux ressources REST API appropriées sur le gestionnaire de files d'attente éloignées:
    - Droit d'insertion de messages dans SYSTEM.ADMIN.COMMAND.QUEUE.
    - Droit d'insertion de messages dans SYSTEM.REST.REPLY.QUEUE.
    - Droit d'accès aux files d'attente de transmission définies pour l'administration à distance.
    - Droit d'affichage des attributs du gestionnaire de files d'attente.
    - Droit d'exécution des demandes REST. Pour plus d'informations, voir la section Security requirements des [rubriques de référence sur les ressources REST API](#).

3. Configurez le gestionnaire de files d'attente local utilisé comme passerelle. Vous pouvez configurer un gestionnaire de files d'attente de passerelle par défaut, spécifier le gestionnaire de files d'attente de passerelle dans un en-tête HTTP ou utiliser une combinaison des deux approches:

- Configurez un gestionnaire de files d'attente de passerelle par défaut à l'aide de la commande **setmqweb** :

```
setmqweb properties -k mqRestGatewayQmgr -v qmgrName
```

où *qmgrName* est le nom du gestionnaire de files d'attente de passerelle.

Ce gestionnaire de files d'attente de passerelle est utilisé lorsque les deux instructions suivantes sont vraies:

- Un gestionnaire de files d'attente n'est pas spécifié dans l'en-tête `ibm-mq-rest-gateway-qmgr` d'une demande REST.
  - Le gestionnaire de files d'attente spécifié dans l'URL de ressource REST API n'est pas un gestionnaire de files d'attente local.
- Configurez le gestionnaire de files d'attente de passerelle sur chaque demande REST en définissant l'en-tête HTTP `ibm-mq-rest-gateway-qmgr` sur le nom du gestionnaire de files d'attente de passerelle.
4. Incluez le nom du gestionnaire de files d'attente éloignées que vous souhaitez administrer dans l'URL de la ressource.

Par exemple, pour obtenir une liste de files d'attente à partir du gestionnaire de files d'attente éloignées `remoteQM`, utilisez l'URL suivante:

```
https://localhost:9443/ibmmq/rest/v1/admin/qmgr/remoteQM/queue
```

## Résultats

Un en-tête `ibm-mq-rest-gateway-qmgr` est renvoyé avec la réponse REST. Cet en-tête indique le gestionnaire de files d'attente qui a été utilisé comme gestionnaire de files d'attente de passerelle.

## Exemple

Dans l'exemple suivant, il existe trois installations IBM MQ sur deux machines. Sous `Machine 1`, il existe un `Installation 1` et un `Installation 2`. Sous `Machine 2`, il existe un `Installation 3`. Un serveur `mqweb` est configuré pour `Installation 1`. Il existe un seul gestionnaire de files d'attente dans chaque installation et ces gestionnaires de files d'attente sont configurés pour l'administration à distance. C'est-à-dire que les programmes d'écoute, les canaux et les files d'attente suivants sont configurés et démarrés:

- Sur le gestionnaire de files d'attente `QM1`, dans `Installation 1`, sous `Machine 1`:
  - Canal émetteur `QM1.to.QM2`
  - Canal récepteur `QM2.to.QM1`
  - Canal émetteur `QM1.to.QM3`
  - Canal récepteur `QM3.to.QM1`
  - File d'attente de transmission `QM2`
  - File d'attente de transmission `QM3`
  - Un programme d'écoute configuré sur le port `1414`
- Sur le gestionnaire de files d'attente `QM2`, dans `Installation 2`, sous `Machine 1`:
  - Canal émetteur `QM2.to.QM1`.
  - Canal récepteur `QM1.to.QM2`
  - File d'attente de transmission `QM1`
  - Un programme d'écoute configuré sur le port `1415`

- Sur le gestionnaire de files d'attente QM3, dans Installation 3, sous Machine 2:
  - Canal émetteur QM3.to.QM1
  - Canal récepteur QM1.to.QM3
  - File d'attente de transmission QM1
  - Le programme d'écoute par défaut

Une file d'attente, Qon2, est définie sur QM2 et une file d'attente Qon3 est définie sur QM3.

L'utilisateur mquser est défini sur les deux machines, reçoit le rôle MQWebAdmin dans REST API et est autorisé à accéder aux files d'attente appropriées sur chaque gestionnaire de files d'attente.

La commande setmqweb permet de configurer le gestionnaire de files d'attente QM1 comme gestionnaire de files d'attente de passerelle par défaut.

Le diagramme suivant illustre cette configuration:

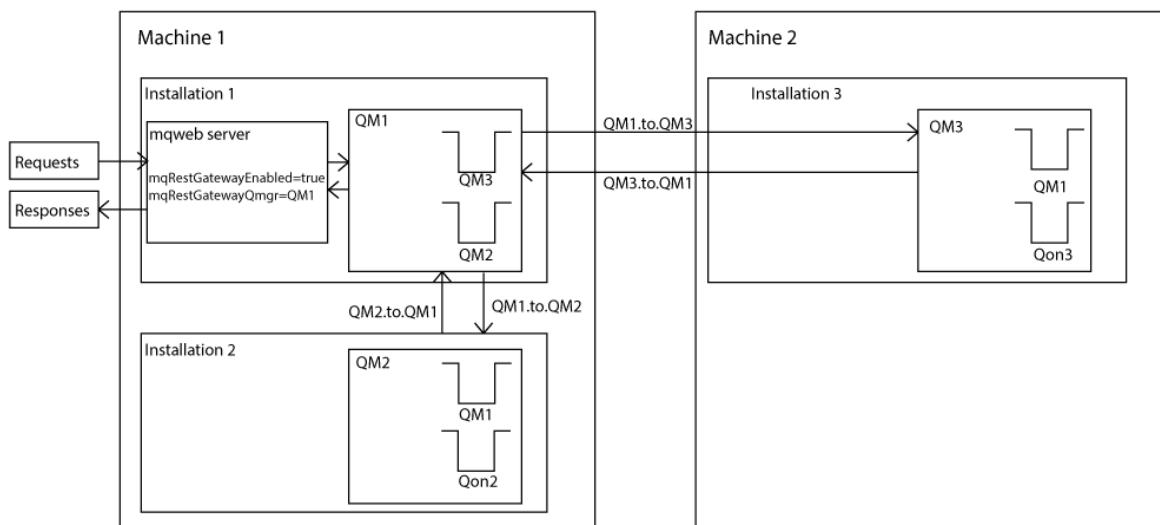


Figure 15. Diagramme d'un exemple de configuration pour l'administration à distance à l'aide de REST API.

La demande REST suivante est envoyée au serveur mqweb:

```
GET https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM2/queue?
attributes=general.isTransmissionQueue
```

La réponse suivante est reçue:

```
{
 "queue" :
 [{
 "general": {
 "isTransmissionQueue": true
 },
 "name": "QM1",
 "type": "local"
 },
 {
 "general": {
 "isTransmissionQueue": false
 },
 "name": "Qon2",
 "type": "local"
 }
]
```

La demande REST suivante est envoyée au serveur mqweb:



```
GET https://localhost:9443/ibmmq/rest/v1/admin/qmgr/QM3/queue?
attributes=general.isTransmissionQueue,general.description
```

La réponse suivante est reçue:

```
{
 "queue" :
 [{
 "general": {
 "isTransmissionQueue": true,
 "description": "Transmission queue for remote admin."
 },
 "name": "QM1",
 "type": "local"
 },
 {
 "general": {
 "isTransmissionQueue": false,
 "description": "A queue on QM3."
 },
 "name": "Qon3",
 "type": "local"
 }
]
```

## V 9.0.2 REST API Horodatages

Lorsque les informations de date et d'heure sont renvoyées par le administrative REST API, elles sont renvoyées en temps universel coordonné (UTC) et dans un format défini.

La date et l'heure sont renvoyées dans le format d'horodatage suivant:

```
YYYY-MM-DDTHH:mm:ss:sssZ
```

Par exemple, 2012-04-23T18:25:43.000Z, où Z indique que le fuseau horaire est le temps universel coordonné (UTC).

Dans IBM MQ 9.0.2, la précision de cet horodatage n'est pas garantie. Par exemple, si le serveur mqweb n'est pas démarré dans le même fuseau horaire que le gestionnaire de files d'attente spécifié dans l'URL de la ressource, l'horodatage peut ne pas être exact. De plus, si des ajustements de l'heure d'été sont nécessaires, l'horodatage peut ne pas être exact.

## REST API traitement des erreurs

Le REST API signale les erreurs en renvoyant un code de réponse HTTP approprié, par exemple 404 (Introuvable) et une réponse JSON. Tout code de réponse HTTP qui n'est pas compris entre 200 et 299 est considéré comme une erreur.

### Format de la réponse d'erreur

La réponse est au format JSON avec le codage UTF-8 . Il contient des objets JSON imbriqués:

- Objet JSON externe qui contient un tableau JSON unique appelé `error`.
- Chaque élément du tableau est un objet JSON qui représente des informations sur une erreur. Chaque objet JSON contient les propriétés suivantes:

#### **type**

Chaîne.

Type d'erreur.

#### **messageId**

Chaîne.

Identificateur unique du message de la forme MQWBnnnnX. Cet identificateur comporte les éléments suivants:

**MQWB**

Préfixe indiquant que le message provient de l'API REST MQ .

**nnnn**

Numéro unique identifiant le message.

**X**

Une seule lettre indiquant la gravité du message:

- I si un message est purement informatif.
- W si un message signale un problème.
- E si un message indique qu'une erreur s'est produite.
- S si un message indique qu'une erreur grave s'est produite.

**message**

Chaîne.

Description de l'erreur.

**explication**

Chaîne.

Explication de l'erreur.

**action**

Chaîne.

Description des étapes pouvant être effectuées pour résoudre l'erreur.

**qmgrName**

**z/OS** Cette zone est disponible uniquement pour z/OS , où le gestionnaire de files d'attente est membre du groupe de partage de files d'attente. Vous devez avoir spécifié le paramètre de requête facultatif **commandScope** ou l'attribut **queueSharingGroupDisposition** .

Chaîne.

Nom du gestionnaire de files d'attente qui a rencontré l'erreur.

**V 9.0.4** Cette zone n'est pas applicable à messaging REST API.

Cette zone n'est disponible que si **type** est pc.f, javaou rest.

Nombre.

Code achèvement MQ associé à l'échec.

**V 9.0.4 reasonCode**

Cette zone n'est disponible que si **type** est pc.f, javaou rest.

Nombre.

Code anomalie MQ associé à l'échec.

**exceptions**

Cette zone n'est disponible que si **type** est java.

Tableau.

Tableau d'exceptions Java ou JMS de chaîne. Chaque élément du tableau des exceptions contient un tableau de chaînes **stackTrace** .

Le tableau de chaînes **stackTrace** contient les détails de chaque exception divisée en lignes.

**Erreurs liées aux groupes de partage de files d'attente****z/OS**

Dans un groupe de partage de files d'attente, il est possible de spécifier un paramètre de requête facultatif **commandScope** pour certaines commandes. Ce paramètre permet de propager la commande à d'autres gestionnaires de files d'attente du groupe de partage de files d'attente. L'une de ces commandes


peut échouer indépendamment, ce qui entraîne la réussite de certaines commandes et l'échec de certaines commandes pour le groupe de partage de files d'attente.

Dans les cas où une commande échoue partiellement, le code d'erreur HTTP 500 est renvoyé. Pour chaque gestionnaire de files d'attente qui a généré un échec, les informations sur cet échec sont renvoyées en tant qu'élément dans le tableau JSON `error`. Pour chaque gestionnaire de files d'attente qui a correctement exécuté la commande, le nom du gestionnaire de files d'attente est renvoyé en tant qu'élément dans un tableau JSON `success`.

## Exemples

- L'exemple suivant illustre la réponse d'erreur à une tentative d'obtention d'informations sur un gestionnaire de files d'attente qui n'existe pas:

```
"error": [
 {
 "type": "rest",
 "messageId": "MQWB0009E",
 "message": "MQWB0009E: Could not query the queue manager 'QM1'",
 "explanation": "The MQ REST API was invoked specifying a queue manager name which cannot be located.",
 "action": "Resubmit the request with a valid queue manager name or no queue manager name, to retrieve a list of queue managers."
 }
]
```

-  L'exemple suivant illustre la réponse d'erreur à une tentative de suppression d'une file d'attente dans un groupe de partage de files d'attente qui n'existe pas pour certains gestionnaires de files d'attente:

```
"error" : [
 {
 "type": "rest",
 "messageId": "MQWB0037E",
 "message": "MQWB0037E: Could not find the queue 'missingQueue' - the queue manager reason code is 3312 : 'MQRCCF_UNKOWNN_OBJECT_NAME'",
 "explanation": "The MQ REST API was invoked specifying a queue name which cannot be located.",
 "action": "Resubmit the request with the name of an existing queue, or with no queue name to retrieve a list of queues.",
 "qmgrName": "QM1"
 },
 {
 "type": "rest",
 "messageId": "MQWB0037E",
 "message": "MQWB0037E: Could not find the queue 'missingQueue' - the queue manager reason code is 3312 : 'MQRCCF_UNKOWNN_OBJECT_NAME'",
 "explanation": "The MQ REST API was invoked specifying a queue name which cannot be located.",
 "action": "Resubmit the request with the name of an existing queue, or with no queue name to retrieve a list of queues.",
 "qmgrName": "QM2"
 }
],
"success" : [{"qmgrName": "QM3"}, {"qmgrName": "QM4"}]
```

## Erreurs liées aux demandes MFT

Si les services MFT REST API ne sont pas activés et que vous appelez le MFT REST API, vous recevez l'exception suivante:

```
{"error": [{
 "action": "Enable the Managed File Transfer REST API and resubmit the request.",
 "completionCode": 0,
 "explanation": "Managed File Transfer REST calls are not permitted as the service is disabled.",
 "message": "MQWB0400E: Managed File Transfer REST API is not enabled.",
 "msgId": "MQWB0400E",
 "reasonCode": 0,
```

```
"type": "rest"
}]}
```

Si les services MFT REST API sont activés et que le gestionnaire de files d'attente de coordination n'est pas défini dans le fichier `mqwebuser.xml`, vous recevez l'exception suivante:

```
{"error": [{
 "action": "Set the coordination queue manager name and restart the mqweb server.",
 "completionCode": 0,
 "explanation": "Coordination queue manager name must be set before using Managed File
Transfer REST services.",
 "message": "MQWB0402E: Coordination queue manager name is not set.",
 "msgId": "MQWB0402E",
 "reasonCode": 0,
 "type": "rest"
}]}
```

## V 9.0.1 REST API découverte

La Documentation de REST API est disponible dans IBM Documentation et au format Swagger. Swagger est une approche couramment utilisée pour documenter les API REST. La documentation Swagger de REST API peut être affichée en activant la fonction de reconnaissance d'API sur le serveur mqweb.

### Avant de commencer

Vous devez activer la sécurité du serveur mqweb pour afficher la documentation Swagger à l'aide de la reconnaissance d'API. Pour plus d'informations sur les étapes requises pour activer la sécurité, voir [Configuration de la sécurité IBM MQ Console](#).

### Procédure

1. Localisez le fichier `mqwebuser.xml` dans l'un des répertoires suivants:

- **ULW** `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **Z/OS** `WLP_user_directory/servers/mqweb`

Où `WLP_user_directory` est le répertoire qui a été spécifié lors de l'exécution du script `crtmqweb.sh` pour créer la définition de serveur mqweb.

2. Ajoutez le code XML approprié au fichier `mqwebuser.xml` :

- Si les balises `<featureManager>` existent dans votre fichier `mqwebuser.xml`, ajoutez le code XML suivant dans les balises `<featureManager>` :
- ```
<feature>apiDiscovery-1.0</feature>
```
- Si les balises `<featureManager>` n'existent pas dans votre fichier `mqwebuser.xml`, ajoutez le code XML suivant dans les balises `<server>` :

```
<featureManager>
  <feature>apiDiscovery-1.0</feature>
</featureManager>
```

3. Affichez la documentation Swagger à l'aide de l'une des méthodes suivantes:

- Affichez une page Web que vous pouvez parcourir et essayer le REST API en entrant l'URL suivante dans un navigateur:

```
https://host:port/ibm/api/explorer
```

Outre l'authentification de chaque demande, vous devez inclure un en-tête `ibm-mq-rest-csrf-token` pour chaque demande POST, PATCH ou DELETE.

V 9.0.5

Le contenu requis de cet en-tête varie en fonction de la version de IBM MQ:

- Depuis la IBM MQ 9.0.5, l'en-tête HTTP `ibm-mq-rest-csrf-token` doit être présent dans la demande ; sa valeur peut être n'importe quelle valeur, y compris vide.
- Avant IBM MQ 9.0.5, la valeur de l'en-tête correspond au contenu du cookie `csrfToken`. Un `csrfToken` est généré lorsqu'une méthode HTTP GET est utilisée avec REST API. Vous pouvez afficher le contenu des cookies en entrant le texte suivant dans la barre d'adresse de votre navigateur:

```
javascript:alert(document.cookie)
```

Cet en-tête de demande est utilisé pour confirmer que les données d'identification utilisées pour authentifier la demande sont utilisées par le propriétaire des données d'identification. C'est-à-dire que le jeton est utilisé pour empêcher les attaques de falsification de requêtes entre sites.

- Extrayez un document Swagger 2 unique qui décrit l'ensemble du REST API en émettant une requête HTTP GET à l'URL suivante:

```
https://host:port/ibm/api/docs
```

Ce document peut être utilisé pour les applications dans lesquelles vous souhaitez naviguer à l'aide d'un programme dans les API disponibles.

host

Indique le nom d'hôte ou l'adresse IP sur lequel REST API est disponible.

La valeur par défaut est `localhost`.

port

Indique le numéro de port HTTPS utilisé par administrative REST API.

La valeur par défaut est `9443`.

Si le nom d'hôte ou le numéro de port par défaut est modifié, vous pouvez déterminer les valeurs correctes à partir de l'URL REST API. Pour plus d'informations sur la détermination de l'URL, voir [«Détermination de l'URL REST API»](#), à la page 76.

V 9.0.4 REST API Support de langue nationale (NLS)

REST API prend en charge, avec certaines qualifications, la possibilité de spécifier des langues nationales dans le cadre d'une demande HTTP.

Arrière-plan

Les en-têtes HTTP permettent de spécifier un comportement particulier sur les demandes et de fournir des informations supplémentaires dans les réponses.

Les en-têtes HTTP incluent la possibilité de demander que les informations soient renvoyées dans une langue nationale. Le REST API respecte cet en-tête dans la mesure du possible.

Spécification d'une langue nationale

Dans l'en-tête HTTP `ACCEPT-LANGUAGE`, une ou plusieurs balises de langue peuvent être fournies. Vous pouvez éventuellement associer un rang aux balises, ce qui permet de spécifier une liste triée par préférence. [Cette page](#) contient une discussion utile sur le principe.

REST API respecte cet en-tête, en sélectionnant une langue dans l'en-tête `ACCEPT-LANGUAGE` et en renvoyant des messages dans cette langue. Lorsque l'en-tête `ACCEPT-LANGUAGE` ne contient aucune langue que le REST API peut prendre en charge, les messages sont renvoyés dans une langue par défaut. Cette langue par défaut correspond à l'environnement local par défaut du serveur Web REST API.

La section [«Quelles données sont traduites?»](#), à la page 94 explique les données qui sont traduites.

Indication de la langue applicable dans les réponses

L'en-tête HTTP CONTENT-LANGUAGE sur les réponses du REST API indique la langue dans laquelle les messages sont renvoyés.

Quelles données sont traduites?

Les messages d'erreur et d'information sont traduits ; les autres textes ne sont pas traduits.

- Les données renvoyées par un gestionnaire de files d'attente ne sont pas traduites. Par exemple, dans le cas de l'exécution d'une commande MQSC via REST API, les réponses du gestionnaire de files d'attente sont dans l'environnement local du gestionnaire de files d'attente.
- La documentation générée (Swagger) pour le REST API, telle qu'exposée via la fonction `apiDiscovery`, est en anglais.

Quelles sont les langues prises en charge?

Outre l'anglais, les messages d'erreur et d'information REST API sont traduits dans les langues suivantes.

Chinois (simplifié)

Dénoté par la balise de langue `zh_CN`

Chinois (traditionnel)

Dénoté par la balise de langue `zh_TW`

tchèque

Dénoté par la balise de langue `cs`

français

Dénoté par la balise de langue `fr`

hongrois

Dénoté par la balise de langue `hu`

italien

Dénoté par la balise de langue `it`

japonais

Dénoté par la balise de langue `ja`

coréen

Dénoté par la balise de langue `ko`

polonais

Dénoté par la balise de langue `pl`

(Brésil) Portugais

Dénoté par la balise de langue `pt_BR`

russe

Dénoté par la balise de langue `ru`

espagnol

Dénoté par la balise de langue `es`

Exemples

Dans les exemples, le serveur Web possède un environnement local par défaut en anglais.

Spécification d'une langue prise en charge unique

Dans les en-têtes de demande, `ACCEPT-LANGUAGE` est défini sur `fr`. Ce paramètre indique que le français est la langue préférée pour le texte traduisible.

Dans les en-têtes de réponse, `CONTENT-LANGUAGE` est défini sur `fr`. Ce paramètre indique que les messages d'erreur et d'information de la réponse sont en français.

Spécification d'une liste de langues

Dans les en-têtes de demande, ACCEPT-LANGUAGE est défini sur `am, fr`. Ce paramètre spécifie que l'amharique et le français sont des langues acceptables pour le texte traduisible et que l'amharique est la langue préférée pour le texte traduisible.

Dans les en-têtes de réponse, CONTENT-LANGUAGE est défini sur `fr`. Ce paramètre indique que les messages d'erreur et d'information de la réponse sont en français, car REST API ne prend pas en charge l'amharique.

Spécification d'une langue unique non prise en charge

Dans les en-têtes de demande, ACCEPT-LANGUAGE est défini sur `am`. Ce paramètre indique que l'amharique est la langue préférée pour le texte traduisible.

Dans les en-têtes de réponse, CONTENT-LANGUAGE est défini sur `en`. Ce paramètre indique que les messages d'erreur et d'information de la réponse sont en anglais, car REST API ne prend pas en charge l'amharique.

V 9.0.1 Administration à l'aide de IBM MQ Console

IBM MQ Console est une interface utilisateur Web qui peut être utilisée pour effectuer des tâches d'administration courantes.

Avant de commencer

Remarque :

- Ne désactivez pas le serveur de commandes sur vos gestionnaires de files d'attente lorsque vous utilisez IBM MQ Console. Si le serveur de commandes est désactivé pour un gestionnaire de files d'attente, IBM MQ Console ne répond plus avec de longs délais au traitement des commandes. Toutes les commandes émises vers le gestionnaire de files d'attente pour lequel le serveur de commandes est désactivé ont un délai d'attente.
- Lorsque vous vous connectez à IBM MQ Console, le navigateur tente d'afficher le IBM MQ Console dans l'environnement local défini par le navigateur. Une vérification est effectuée pour voir si IBM MQ Console prend en charge la langue spécifiée. Si les fichiers de langue sont introuvables, l'anglais américain est utilisé et le serveur mqweb les consigne en tant qu'erreurs de fichier introuvable. Par conséquent, les messages suivants sont attendus lorsque votre navigateur est défini sur une langue qui n'est pas prise en charge par IBM MQ Console:

```
SRVE0190E: File not found: /nls/en_GB/labels.json
SRVE0190E: File not found: /nls/en_GB/pcf.json
SRVE0190E: File not found: /nls/en_GB/errors.json
SRVE0190E: File not found: /nls/en_GB/strings.json
```

Procédure

- [Utilisation des gestionnaires de files d'attente locales](#)
- [Utilisation des objets IBM MQ](#)
- [Utilisation d'enregistrements de droits d'accès](#)
- [Surveillance de l'utilisation des ressources système](#)
- [Configuration des présentations de tableau de bord](#)

V 9.0.1 Guide d'initiation à IBM MQ Console

Avant de démarrer le IBM MQ Console, vous devez installer les composants appropriés et démarrer le serveur mqweb. Vous pouvez ensuite démarrer le IBM MQ Console dans un navigateur.

Avant de commencer

IBM i

Sous IBM i, les commandes doivent être exécutées dans QSHLL.

Pourquoi et quand exécuter cette tâche

La procédure de cette tâche se concentre sur les étapes de base pour démarrer rapidement avec IBM MQ Console. Pour plus d'informations sur la personnalisation de votre configuration, voir les liens qui sont répertoriés sous l'étape suivante.

Remarque : Vous avez la possibilité de configurer IBM MQ Console sans sécurité lors de l'installation.

Procédure

1. Installez le composant IBM MQ Console et REST API :

- **V 9.0.4** **AIX** Sous AIX, installez l'ensemble de fichiers `mqm.web.rte`.
- **Linux** Sous Linux, installez le composant `MQSeriesWeb`. Pour plus d'informations sur l'installation de composants et de fonctions sous Linux, voir [Tâches d'installation d'Linux](#).
- **Windows** Sous Windows, installez la fonction `Web Administration`. Pour plus d'informations sur l'installation de composants et de fonctions sous Windows, voir [Tâches d'installation d'Windows](#).
- **z/OS** Sous z/OS, installez la fonction `IBM MQ for z/OS Unix System Web Services Components`. Pour plus d'informations sur l'installation de composants et de fonctions sous z/OS, voir [Tâches d'installation d'z/OS](#).

2. Configurez la sécurité de base pour autoriser les utilisateurs et les groupes à accéder au IBM MQ Console.

a) Copiez l'exemple de fichier `basic_registry.xml` du répertoire `MQ_INSTALLATION_PATH/web/mq/samp/configuration` dans:

- **ULW** Sous UNIX, Linux, and Windows: `MQ_DATA_DIRECTORY/web/installations/installationName/servers/mqweb`
- **z/OS** Sous z/OS: `WLP_user_directory/servers/mqweb`
où `WLP_user_directory` est le répertoire qui a été spécifié lors de l'exécution du script `crtmqweb.sh` pour créer la définition du serveur `mqweb`.

b) Renommez l'exemple de fichier XML en `mqwebuser.xml`.

Remarque : Ce fichier renommé remplace un fichier existant qui est également utilisé pour administrative REST API. Par conséquent, si vous avez modifié le fichier `mqwebuser.xml` pour administrative REST API, copiez vos modifications dans le nouveau fichier XML avant de le renommer.

3. En fonction de la plateforme, activez les connexions distantes au serveur `mqweb`:

- **Linux** Facultatif sur Linux
- **Windows** Facultatif sur Windows
- **z/OS** Sous z/OS
- Depuis IBM MQ 9.0.4, utilisez la commande **setmqweb** :
`setmqweb properties -k httpHost -v hostname`
- Depuis IBM MQ 9.0.1, ajoutez le code XML suivant au fichier `mqwebuser.xml`, dans les balises `<server>` :
`<variable name="httpHost" value="hostname"/>`

où `nom_hôte` indique l'adresse IP, le nom d'hôte du serveur de noms de domaine (DNS) avec le suffixe du nom de domaine ou le nom d'hôte DNS du serveur sur lequel IBM MQ est installé. Utilisez un astérisque (*) pour indiquer toutes les interfaces réseau disponibles.



Avertissement : z/OS V 9.0.4

Avant d'émettre la commande **setmqweb** ou **dspmqweb** sous z/OS, vous devez définir la variable d'environnement `WLP_USER_DIR` de sorte qu'elle pointe vers votre configuration de serveur mqweb.

Pour ce faire, entrez la commande suivante :

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb.sh`. Exemple :

```
export WLP_USER_DIR=/var/mqm/web/installation1
```

Pour plus d'informations, voir [Création de la définition de serveur Liberty](#).

4. Démarrez le serveur mqweb qui prend en charge le IBM MQ Console:

- Windows Linux Sous Windows et Linux, en tant qu' utilisateur privilégié, entrez la commande suivante sur la ligne de commande:
`strmqweb`
- z/OS Sous z/OS, démarrez la procédure que vous avez créée dans [Tâche 29: Créer une procédure pour le IBM WLP](#).

5. Connectez-vous à IBM MQ Console en entrant l'URL suivante dans un navigateur:

```
https://hostname:9443/ibmmq/console
```

où `nom_hôte` indique l'adresse IP ou le nom d'hôte du serveur de noms de domaine (DNS) avec le suffixe du nom de domaine ou le nom d'hôte DNS du serveur sur lequel IBM MQ est installé. Si vous n'avez pas configuré de connexions distantes à l'étape 3, la valeur de `hostname` est `localhost`.

6. Connectez vous au IBM MQ Console. A partir de IBM MQ 9.0.2, utilisez le nom d'utilisateur `mqadminet` le mot de passe `mqadmin`. Dans IBM MQ 9.0.1, utilisez le nom d'utilisateur `adminet` le mot de passe `admin`.

Que faire ensuite

- Pour plus d'informations sur la configuration de la sécurité IBM MQ Console , notamment sur la manière de configurer les utilisateurs et les groupes, LDAP et les certificats client, voir [Configuration de la sécurité IBM MQ Console](#).
- Pour plus d'informations sur la configuration des paramètres IBM MQ Console , y compris l'activation des connexions HTTP, voir [Configuration de la console IBM MQ](#).
- Pour plus d'informations sur la détermination de l'URL s'il ne s'agit pas de l'URL par défaut, voir «Détermination de l'URL IBM MQ Console», à la page 97.

V 9.0.1

Détermination de l'URL IBM MQ Console

L'URL par défaut permettant d'accéder au IBM MQ Console est `https://localhost:9443/ibmmq/console`. Si l'hôte ou le port HTTP par défaut est modifié ou si le port HTTP est activé, vous pouvez déterminer l'URL à l'aide de la commande **dspmqweb** .

Pourquoi et quand exécuter cette tâche

V 9.0.4

Depuis la IBM MQ 9.0.4, vous pouvez utiliser la commande **dspmqweb status** pour déterminer l'URL IBM MQ Console sous Windows, Linux et z/OS. Pour IBM MQ 9.0.3 et les versions antérieures, sous Windows et Linux, vous pouvez utiliser la commande **dspmqweb** . Sous z/OS , vous devez effectuer une recherche dans le fichier `messages.log` pour déterminer l'URL.

Procédure

V 9.0.4

Utilisez l'une des méthodes suivantes pour déterminer l'URL:

- Depuis IBM MQ 9.0.4, utilisez la commande **dspmqweb status** en tant qu' utilisateur privilégié:
 1. Vérifiez que le serveur mqweb est en cours d'exécution en entrant la commande **strmqweb** sur la ligne de commande.
 2. Affichez l'URL en entrant la commande `dspmqweb status` sur la ligne de commande.
- Pour IBM MQ 9.0.3 et les versions antérieures, utilisez l'une des méthodes suivantes:

- **Windows** > **Linux** Sous Windows ou Linux, utilisez la commande **dspmqweb** en tant qu' utilisateur privilégié:

1. Vérifiez que le serveur mqweb est en cours d'exécution en entrant la commande **strmqweb** sur la ligne de commande.
2. Affichez l'URL en entrant la commande `dspmqweb` sur la ligne de commande.

- **z/OS** Sous z/OS, utilisez le fichier messages .log pour rechercher l'URL:

1. Ouvrez le fichier messages .log.

Le fichier messages .log se trouve dans le chemin `WLP_user_directory/servers/mqweb/logs`, où `WLP_user_directory` est le répertoire qui a été spécifié lors de l'exécution du script **crtmqweb.sh** pour créer la définition de serveur mqweb.

2. Recherchez le dernier code de message CWWKT0016I qui se termine par `ibmmq/console`. L'URL est incluse dans ce message.

Remarque : IBM MQ Console utilise l'URL interne `/ibmmq/console/internal`. Cette URL est affichée dans certains messages CWWKT0016I, mais elle ne doit être utilisée que par le IBM MQ Console.

Exemple

L'exemple suivant illustre le code de message CWWKT0016I avec l'URL comme URL par défaut:

```
A CWWKT0016I: Web application available (default_host): https://localhost:9443/ibmmq/console
```

V 9.0.1 > z/OS Restrictions sur z/OS

Les restrictions suivantes s'appliquent lorsque vous utilisez IBM MQ Console pour gérer les gestionnaires de files d'attente sous z/OS.

- Les gestionnaires de files d'attente sur z/OS ne peuvent pas être créés, supprimés, démarrés ou arrêtés.
- Les initiateurs de canal sous z/OS ne peuvent pas être démarrés ou arrêtés et le statut de l'initiateur de canal n'est pas affiché.
- Les programmes d'écoute ne peuvent pas être affichés ou administrés.
- Les commandes de canal de démarrage, de commande ping, de résolution et de réinitialisation ne peuvent être émises qu'avec CHLDISP (DEFAULT).
- Les nouveaux objets ne peuvent être créés qu'avec QSGDISP (QMGR).
- Les objets définis avec QSGDISP (GROUP) ne peuvent pas être affichés ou gérés.
- La sécurité du gestionnaire de files d'attente ne peut pas être gérée.
- L'utilisation des ressources système ne peut pas être surveillée.

Tâches associées

«Administration à l'aide de IBM MQ Console», à la page 95

IBM MQ Console est une interface utilisateur Web qui peut être utilisée pour effectuer des tâches d'administration courantes.

Information associée

[Utilisation des gestionnaires de files d'attente locales](#)

V 9.0.1 Utilisation des gestionnaires de files d'attente locales

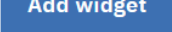
Vous pouvez utiliser le widget de gestionnaire de files d'attente local dans le IBM MQ Console pour créer, configurer et contrôler des gestionnaires de files d'attente locaux.

Pourquoi et quand exécuter cette tâche

Le widget de gestionnaire de files d'attente local répertorie les gestionnaires de files d'attente locaux qui sont ajoutés à l'installation IBM MQ à partir de laquelle IBM MQ Console s'exécute. Les gestionnaires de files d'attente qui sont associés à des installations différentes d' IBM MQ sur le même système ne sont pas répertoriés. Vous pouvez sélectionner des gestionnaires de files d'attente individuels dans la liste.

Remarque : V 9.0.4 IBM MQ Console ne prend pas en charge les gestionnaires de files d'attente de données répliquées (RDQM).

Vous pouvez ajouter un widget de gestionnaire de files d'attente local à votre tableau de bord en

cliquant sur **Ajouter un widget** V 9.0.5 . Sélectionnez ensuite **Gestionnaires de files d'attente locales**.



V 9.0.5 Vous pouvez renommer le widget après l'avoir créé. Placez le pointeur de la souris dans la






barre de titre et l'icône de changement de nom apparaît. Cliquez sur l'icône, entrez un nouveau nom dans la fenêtre du widget Renommer, puis cliquez sur **Renommer**.

z/OS Vous ne pouvez pas créer, démarrer, arrêter ou supprimer des gestionnaires de files d'attente sur z/OS.

Procédure

- Pour créer un gestionnaire de files d'attente local:
 - a) Cliquez sur l'icône de création V 9.0.5  dans la barre d'outils du widget de gestionnaire de files d'attente local.
 - b) Entrez un nom pour le nouveau gestionnaire de files d'attente. Vous pouvez indiquer 48 caractères au maximum. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_", et "%".
 - c) Facultatif : Entrez un port TCP/IP disponible sur lequel le gestionnaire de files d'attente sera à l'écoute. Le numéro de port ne doit pas dépasser 65535.
 - d) Cliquez sur **Create**. Le nouveau gestionnaire de files d'attente est créé et démarré.
- Pour démarrer un gestionnaire de files d'attente local:
 - a) Sélectionnez le gestionnaire de files d'attente à démarrer dans la liste du widget de gestionnaire de files d'attente local.
 - b) Cliquez sur l'icône de démarrage V 9.0.5  dans la barre d'outils du widget de gestionnaire de files d'attente local.
- Pour arrêter un gestionnaire de files d'attente local:

- a) Sélectionnez le gestionnaire de files d'attente à arrêter dans la liste du widget de gestionnaire de files d'attente local.
 - b) Cliquez sur l'icône d'arrêt  dans la barre d'outils du widget de gestionnaire de files d'attente local.
 - c) Confirmez l'arrêt du gestionnaire de files d'attente en cliquant sur **Arrêter**.
- Pour supprimer un gestionnaire de files d'attente local:
 - a) Sélectionnez le gestionnaire de files d'attente à supprimer dans la liste du widget de gestionnaire de files d'attente local.
 - b) Si le gestionnaire de files d'attente est en cours d'exécution, arrêtez-le.
 - c) Cliquez sur l'icône de suppression  dans la barre d'outils du widget de gestionnaire de files d'attente local.
 - d) Confirmez la suppression du gestionnaire de files d'attente en cliquant sur **Supprimer**. Le gestionnaire de files d'attente et tous les objets associés sont supprimés.
 - Pour afficher et éditer les propriétés d'un gestionnaire de files d'attente local:
 - a) Assurez-vous que le gestionnaire de files d'attente est en cours d'exécution et sélectionnez-le dans la liste des gestionnaires de files d'attente.
 - b) Cliquez sur l'icône des propriétés  dans la barre d'outils du widget de gestionnaire de files d'attente local. Vous pouvez également cliquer deux fois sur le gestionnaire de files d'attente.
 - c) Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande. Pour plus d'informations sur une propriété, voir [Propriétés du gestionnaire de files d'attente](#) dans la documentation de MQ Explorer.
 - Pour actualiser la sécurité du gestionnaire de files d'attente local:
 - a) Vérifiez que le gestionnaire de files d'attente local est en cours d'exécution et sélectionnez-le dans la liste des gestionnaires de files d'attente.
 - b) Sélectionnez ... > **Actualiser la sécurité**
 - c) Sélectionnez la sécurité du gestionnaire de files d'attente à actualiser:
 - Sélectionnez **Service d'autorisation** pour actualiser la liste des autorisations détenues en interne par le composant des services d'autorisation.
 - Sélectionnez **Authentification de connexion** pour actualiser la vue en cache de la configuration de l'authentification de connexion.
 - Sélectionnez **SSL** pour actualiser la vue en cache du référentiel de clés SSL ou TLS. Cette option actualise également les emplacements des serveurs LDAP utilisés pour les listes de révocation certifiées, ainsi que les paramètres matériels de chiffrement.
 - Pour utiliser des enregistrements de droits d'accès pour le gestionnaire de files d'attente local:
 - a) Vérifiez que le gestionnaire de files d'attente local est en cours d'exécution et sélectionnez-le dans la liste des gestionnaires de files d'attente.
 - b) Sélectionnez l'une des options ci-après :
 - Sélectionnez ... > **Gestion des enregistrements de droits d'accès** pour utiliser les enregistrements de droits d'accès du gestionnaire de files d'attente et spécifier les actions que les groupes d'utilisateurs peuvent effectuer.
 - Sélectionnez ... > **Gestion des enregistrements de droits de création** pour utiliser les enregistrements de droits de création pour le gestionnaire de files d'attente et spécifier les objets que les groupes d'utilisateurs peuvent créer sur ce gestionnaire de files d'attente.
 - Pour créer automatiquement un onglet de tableau de bord pour un gestionnaire de files d'attente local:

- a) Sélectionnez le gestionnaire de files d'attente dans le widget de gestionnaire de files d'attente local.
- b) Sélectionnez ... > **Ajouter un nouvel onglet de tableau de bord**
Un nouvel onglet de tableau de bord est créé. L'onglet contient le nom du gestionnaire de files d'attente.
- Pour filtrer la liste des gestionnaires de files d'attente locaux:
 - a) Entrez le texte de votre filtre dans la zone de recherche.
 - b) Pour arrêter le filtrage, supprimez le texte de la zone de recherche.

V 9.0.1 Utilisation des objets IBM MQ

Vous pouvez utiliser les widgets d'objet IBM MQ dans le IBM MQ Console pour utiliser les différents types d'objet IBM MQ .


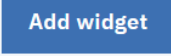
Pourquoi et quand exécuter cette tâche

Chaque widget d'objet IBM MQ contient des objets associés à un gestionnaire de files d'attente spécifique. Vous pouvez ajouter les types suivants de widgets d'objet IBM MQ à votre tableau de bord:

- Widget Files d'attente
- Widget Rubriques
- Widget Programmes d'écoute
- Widget Canaux
- Widget Canaux de connexion client
- Widget Informations d'authentification
- Widget Abonnements

Procédure

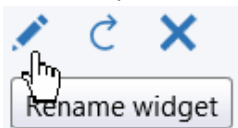
- Pour créer un widget d'objet IBM MQ :

- a) Cliquez sur **Ajouter un widget**  .
- b) Sélectionnez le gestionnaire de files d'attente approprié dans la liste.
- c) Cliquez sur le nom du type de widget d'objet que vous souhaitez créer.

- 

Pour renommer le widget d'objet:

- a) Placez le pointeur de la souris dans la barre de titre et l'icône de changement de nom apparaît






. Cliquez sur l'icône.

- b) Entrez un nouveau nom dans la fenêtre du widget Renommer et cliquez sur **Renommer**.

- 

Pour configurer le widget d'objet IBM MQ :

- a) Cliquez sur l'icône de configuration  dans la barre de titre du widget.
- b) Facultatif : Indiquez le gestionnaire de files d'attente pour lequel les objets IBM MQ sont affichés.
- c) Facultatif : Indiquez si les objets système sont affichés ou masqués.
- d) Cliquer sur **Sauvegarder**.

- Pour filtrer les objets affichés dans le widget:
 - a) Entrez le texte de votre filtre dans la zone de recherche.
 - b) Pour arrêter le filtrage, supprimez le texte de la zone de recherche.
- Pour actualiser le contenu du widget, cliquez sur l'icône Actualiser  dans la barre de titre du widget.
- Pour supprimer le widget, cliquez sur l'icône de suppression  dans la barre de titre du widget.

Utilisation des files d'attente


Vous pouvez utiliser le widget Files d'attente dans IBM MQ Console pour afficher les files d'attente qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez ensuite ajouter et supprimer des files d'attente, ajouter et supprimer des messages dans une file d'attente, parcourir des messages, afficher et définir les propriétés d'une file d'attente et gérer les enregistrements de droits d'accès d'une file d'attente.

Avant de commencer


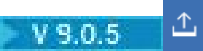

Vous devez créer un widget de files d'attente avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir «Utilisation des objets IBM MQ», à la page 101.

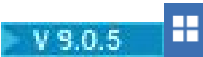



Pourquoi et quand exécuter cette tâche


Le widget des files d'attente répertorie les files d'attente qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez sélectionner des files d'attente individuelles dans la liste à gérer.

 Vous ne pouvez pas afficher ni éditer les enregistrements de droits d'accès pour les files d'attente sur z/OS.

Procédure

- Pour ajouter une file d'attente :
 - a) Cliquez sur l'icône Créer  dans la barre d'outils du widget Files d'attente.
 - b) Entrez un nom pour la file d'attente. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_ et %".
 - c) Sélectionnez le type de file d'attente à ajouter.
 - d) Cliquez sur **Create**. La nouvelle file d'attente est créée.
- Pour placer des messages dans une file d'attente:
 - a) Sélectionnez la file d'attente à laquelle vous souhaitez ajouter des messages dans la liste du widget des files d'attente. Vous ne pouvez pas sélectionner de file d'attente modèle.
 - b) Cliquez sur l'icône d'insertion de message  dans la barre d'outils du widget Files d'attente.
 - c) Entrez le message que vous souhaitez placer dans la file d'attente.
 - d) Cliquez sur **Insertion**.
- Pour effacer des messages dans une file d'attente :
 - a) Sélectionnez la file d'attente locale dans laquelle vous souhaitez effacer les messages dans la liste du widget des files d'attente.
 - b) Sélectionnez  ... > **File d'attente de suppression**.

- c) Confirmez l'effacement de la file d'attente en cliquant sur **Effacer la file d'attente**.
- Pour parcourir les messages d'une file d'attente:
 - a) Sélectionnez la file d'attente locale ou alias à parcourir dans la liste du widget des files d'attente.
 - b) Cliquez sur l'icône Parcourir  dans la barre d'outils du widget Files d'attente. La fenêtre Exploration des messages s'ouvre et affiche les messages de la file d'attente.
- Pour supprimer une file d'attente :
 - a) Sélectionnez la file d'attente à supprimer dans la liste du widget de files d'attente.
 - b) Cliquez sur l'icône Supprimer  dans la barre d'outils du widget Files d'attente.
 - c) Facultatif : Si la file d'attente contient des messages, confirmez qu'elle peut être effacée en cliquant sur **Mettre à blanc la file d'attente**.
 - d) Confirmez la suppression de la file d'attente en cliquant sur **Supprimer**. La file d'attente est supprimée.
- Pour afficher et éditer les propriétés d'une file d'attente :
 - a) Sélectionnez la file d'attente dans le widget de files d'attente.
 - b) Cliquez sur l'icône Propriétés  dans la barre d'outils du widget Files d'attente. Vous pouvez également cliquer deux fois sur la file d'attente.
 - c) 

Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande. Pour plus d'informations sur une propriété, consultez les informations de propriété dans [Propriétés de file d'attente](#) dans la documentation MQ Explorer.
- Afin d'afficher et d'éditer des enregistrements de droits d'accès pour une file d'attente :
 - a) Sélectionnez la file d'attente dans le widget.
 - b) Cliquez sur  **> Gérer les enregistrements de droits d'accès**.
Les enregistrements de droits d'accès affichent les droits dont les utilisateurs et les administrateurs disposent pour la file d'attente sélectionnée.

Utilisation des rubriques


Vous pouvez utiliser le widget de rubriques dans le IBM MQ Console pour ajouter et supprimer des rubriques, ainsi que pour afficher et définir les propriétés d'une rubrique.

Avant de commencer

Vous devez créer un widget de rubriques avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir [«Utilisation des objets IBM MQ»](#), à la page 101.













Pourquoi et quand exécuter cette tâche

Le widget de rubriques répertorie les rubriques qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez sélectionner des rubriques individuelles dans la liste à utiliser.

 Vous ne pouvez pas afficher ni éditer les enregistrements de droits d'accès d'une rubrique sur z/OS.

Procédure

- Pour ajouter une rubrique :

- a) Cliquez sur l'icône de création   dans la barre d'outils du widget de rubriques.
- b) Entrez un nom pour la nouvelle rubrique. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ":", "/", "_ " et "%".
- c) Spécifiez la chaîne de rubrique dans laquelle vous publiez des messages pour la rubrique. Pour plus d'informations, voir [Propriétés des rubriques](#).
- d) Cliquez sur **Create**. La nouvelle rubrique est créée.
- Pour supprimer une rubrique :
 - a) Sélectionnez la rubrique à supprimer dans la liste du widget de rubriques.
 - b) Cliquez sur l'icône de suppression   dans la barre d'outils du widget de rubriques.
 - c) Confirmez la suppression de la rubrique en cliquant sur **Supprimer**. La rubrique est supprimée.
- Pour afficher et éditer les propriétés d'une rubrique :
 - a) Sélectionnez la rubrique dans le widget de rubriques.
 - b) Cliquez sur l'icône des propriétés   dans la barre d'outils du widget de rubriques. Vous pouvez également cliquer deux fois sur la rubrique.
 - c)  Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande. Pour plus d'informations sur une propriété, consultez les informations sur les propriétés dans [Propriétés des rubriques](#) dans la MQ Explorer.
- Pour publier un message sur une rubrique:
 - a) Cliquez sur l'icône Insérer un message   dans la barre d'outils du widget de rubriques.
 - b) Entrez un message dans la zone **Message** .
 - c) Entrez la chaîne de rubrique sur laquelle publier le message dans la zone **Chaîne de rubrique** .
 - d) Cliquez sur **Publication**.
- Pour s'abonner à une rubrique :
 - a) Cliquez sur l'icône d'abonnement   dans la barre d'outils du widget de rubriques.
 - b) Entrez la chaîne de rubrique à laquelle vous souhaitez vous abonner dans la zone **Chaîne de rubrique** .
 - c) Cliquez sur **Subscribe**.
- Afin d'afficher et d'éditer des enregistrements de droits d'accès pour une rubrique :
 - a) Sélectionnez la rubrique dans le widget de rubriques.
 - b) Cliquez sur le  ... > **Gérer les enregistrements de droits d'accès**.
Les enregistrements de droits d'accès affichent les droits dont les utilisateurs et les administrateurs disposent pour la rubrique sélectionnée.

Utilisation de programmes d'écoute


Vous pouvez utiliser le widget des programmes d'écoute dans le IBM MQ Console pour ajouter et supprimer des programmes d'écoute, démarrer et arrêter des programmes d'écoute, afficher et définir des propriétés de programme d'écoute et gérer les enregistrements de droits d'accès d'un programme d'écoute.

Avant de commencer







Vous devez créer un widget d'écoute avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir «Utilisation des objets IBM MQ», à la page 101.


Pourquoi et quand exécuter cette tâche

Le widget Programmes d'écoute répertorie les programmes d'écoute qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez sélectionner des programmes d'écoute individuels dans la liste pour les utiliser.

 Vous ne pouvez pas utiliser le widget d'écoute sur z/OS.

Procédure

- Pour ajouter un programme d'écoute TCP/IP:
 - a) Cliquez sur l'icône de création  dans la barre d'outils du widget d'écoute.
 - b) Entrez un nom pour le programme d'écoute. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_" et "%".
 - c) Entrez un port TCP/IP disponible pour le programme d'écoute. Le numéro de port ne doit pas dépasser 65535.
 - d) Cliquez sur **Create**. Le nouveau programme d'écoute est créé.
- Pour supprimer un programme d'écoute :
 - a) Sélectionnez le programme d'écoute à supprimer dans la liste du widget de programmes d'écoute.
 - b) Cliquez sur l'icône de suppression  dans la barre d'outils du widget d'écoute.
 - c) Confirmez la suppression du programme d'écoute en cliquant sur **Supprimer**. Le programme d'écoute est supprimé.
- Pour démarrer un programme d'écoute :
 - a) Sélectionnez le programme d'écoute à démarrer dans la liste du widget des programmes d'écoute.
 - b) Cliquez sur l'icône de démarrage  dans la barre d'outils du widget d'écoute.
- Pour arrêter un programme d'écoute :
 - a) Sélectionnez le programme d'écoute à arrêter dans la liste du widget des programmes d'écoute.
 - b) Cliquez sur l'icône d'arrêt  dans la barre d'outils du widget d'écoute.
 - c) Confirmez l'arrêt du programme d'écoute en cliquant sur **Arrêter**.
- Pour afficher et éditer les propriétés d'un programme d'écoute :
 - a) Sélectionnez le programme d'écoute dans le widget Programmes d'écoute.
 - b) Cliquez sur l'icône Propriétés  dans la barre d'outils du widget d'écoute. Vous pouvez également cliquer deux fois sur le programme d'écoute.
 - c) 
Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande. Pour plus d'informations sur une propriété, consultez les informations de propriété dans [Propriétés du programme d'écoute](#) dans la MQ Explorer.
- Afin d'afficher et d'éditer des enregistrements de droits d'accès pour un programme d'écoute :
 - a) Sélectionnez le programme d'écoute dans le widget Programmes d'écoute.

- b) Cliquez sur  ... > **Gérer les enregistrements de droits d'accès**. Les enregistrements de droits d'accès affichent les droits dont les utilisateurs et les administrateurs disposent pour le programme d'écoute sélectionné.

Utilisation des canaux


Vous pouvez utiliser le widget de canaux dans le IBM MQ Console pour ajouter et supprimer des canaux, démarrer et arrêter des canaux, réinitialiser et résoudre des canaux et des canaux ping. Vous pouvez également afficher et définir les propriétés d'un canal et gérer les enregistrements de droits d'accès pour le canal.

Avant de commencer









Vous devez créer un widget de canaux avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir «Utilisation des objets IBM MQ», à la page 101.



Pourquoi et quand exécuter cette tâche


Le widget Canaux répertorie les canaux qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez sélectionner des canaux individuels dans la liste pour les utiliser.




 Vous ne pouvez pas afficher ni éditer les enregistrements de droits d'accès d'un canal sur z/OS.

Procédure

- Pour ajouter un canal :
 - a) Cliquez sur l'icône Créer  **Create**  dans la barre d'outils du widget Canaux.
 - b) Entrez un nom pour le canal. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_", et "%".
 - c) Sélectionnez le type de canal à ajouter.
 - d) Si vous créez un canal émetteur, émetteur de cluster ou demandeur, spécifiez le nom de connexion. Il s'agit du nom de l'ordinateur qui héberge le gestionnaire de files d'attente cible. Le format du nom est *nom_ordinateur(numéro_port)*. *nom_ordinateur* est le nom ou l'adresse IP de l'ordinateur qui héberge le gestionnaire de files d'attente cible et *numéro_port* est le port utilisé par le programme d'écoute du gestionnaire de files d'attente cible.
 - e) Si vous créez un canal émetteur ou un canal serveur, spécifiez le nom de la file d'attente de transmission qui correspond au gestionnaire de files d'attente à l'extrémité réceptrice du canal.
 - f) Cliquez sur **Create**. Le nouveau canal est créé.
- Pour supprimer un canal :
 - a) Sélectionnez le canal à supprimer dans la liste du widget Canaux.
 - b) Cliquez sur l'icône de suppression   dans la barre d'outils du widget.
 - c) Confirmez la suppression du canal en cliquant sur **Supprimer**. Le canal est supprimé.
- Pour démarrer un canal, procédez comme suit :
 - a) Sélectionnez le canal à démarrer dans la liste du widget Canaux.
 - b) Cliquez sur l'icône de démarrage   dans la barre d'outils du widget.
- Pour arrêter un canal, procédez comme suit :
 - a) Sélectionnez le canal à arrêter dans la liste du widget Canaux.
 - b) Cliquez sur l'icône d'arrêt   dans la barre d'outils du widget.

- c) Confirmez l'arrêt du canal en cliquant sur **Arrêter**.
- Pour afficher les propriétés d'un canal :
 - a) Sélectionnez le canal dans le widget Canaux.
 - b) Cliquez sur l'icône Propriétés  dans la barre d'outils du widget Canaux. Vous pouvez également cliquer deux fois sur le canal.
 - c) 

Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande. Pour plus d'informations sur une propriété, consultez les informations de propriété dans [Propriétés du canal](#) dans la documentation de MQ Explorer.
- Pour réinitialiser un canal :
 - a) Sélectionnez le canal dans le widget Canaux.
 - b) Cliquez sur  ... > **Réinitialiser**.
 - c) Spécifiez un numéro de séquence de message.

Vous devez réinitialiser un canal s'il ne démarre pas car les deux extrémités divergent sur le numéro de message du message suivant à envoyer. Le numéro de séquence de message spécifie ce numéro.
 - d) Cliquez sur **Réinitialiser le canal**.
- Pour résoudre un canal, procédez comme suit :
 - a) Sélectionnez le canal dans le widget Canaux.
 - b) Cliquez sur  ... > **Résoudre**.
 - c) Choisissez de valider ou d'éliminer le lot de messages en cours en cliquant sur **Valider** ou sur **Annulation**.
- Pour effectuer le test Ping d'un canal, procédez comme suit :
 - a) Sélectionnez le canal dans le widget Canaux.
 - b) Cliquez sur  ... > **Ping**.
- Afin d'afficher ou éditer des enregistrements de droits d'accès pour un canal :
 - a) Sélectionnez le canal dans le widget.
 - b) Cliquez sur  ... > **Gérer les enregistrements de droits d'accès**.

Les enregistrements de droits d'accès affichent les droits dont les utilisateurs et les administrateurs disposent pour le canal sélectionné.

Utilisation de canaux de connexion client


Vous pouvez utiliser le widget Canaux de connexion client dans le IBM MQ Console pour ajouter et supprimer des canaux de connexion client sur un gestionnaire de files d'attente, afficher et définir les propriétés et gérer les enregistrements de droits d'accès pour le canal.

Avant de commencer








Vous devez créer un widget de canaux de connexion client avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir [«Utilisation des objets IBM MQ»](#), à la page 101.


Pourquoi et quand exécuter cette tâche

Le widget Canaux de connexion client répertorie les canaux de connexion client qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez sélectionner des canaux de connexion client individuels dans la liste à utiliser.

 Vous ne pouvez pas afficher ni éditer les enregistrements de droits d'accès d'un canal de connexion client sur z/OS.

Procédure

- Pour ajouter un canal de connexion client :
 - a) Cliquez sur l'icône Créer   dans la barre d'outils du widget Canaux de connexion client.
 - b) Entrez un nom pour le canal de connexion client. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_" et "%".
 - c) Spécifiez le nom de la connexion. Il s'agit du nom de l'ordinateur qui héberge le gestionnaire de files d'attente cible. Le format est *nom_ordinateur(numéro_port)*, où *nom_ordinateur* est le nom ou l'adresse IP de l'ordinateur qui héberge le gestionnaire de files d'attente cible, et *numéro_port* est le port utilisé par le programme d'écoute du gestionnaire de files d'attente cible.
 - d) Cliquez sur **Create**. Le nouveau canal de connexion client est créé.
- Pour supprimer un canal de connexion client :
 - a) Sélectionnez le canal de connexion client à supprimer dans la liste du widget Canaux de connexion client.
 - b) Cliquez sur l'icône de suppression   dans la barre d'outils du widget.
 - c) Confirmez la suppression du canal de connexion client en cliquant sur **Supprimer**. Le canal de connexion client est supprimé.
- Pour afficher et éditer les propriétés d'un canal de connexion client :
 - a) Sélectionnez le canal de connexion client dans le widget des canaux de connexion client.
 - b) Cliquez sur l'icône Propriétés   dans la barre d'outils du widget Canaux de connexion client. Vous pouvez également cliquer deux fois sur le canal de connexion client.
 - c) 

Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande. Pour plus d'informations sur une propriété, consultez les informations de propriété dans [Propriétés du canal](#) dans la documentation de MQ Explorer.
- Afin d'afficher et d'éditer des enregistrements de droits d'accès pour un canal de connexion client :
 - a) Sélectionnez le canal de connexion client dans le widget des canaux de connexion client.
 - b) Cliquez sur  ... > **Gérer les enregistrements de droits d'accès**. Les enregistrements de droits d'accès affichent les droits dont les utilisateurs et les administrateurs disposent pour le canal de connexion client sélectionné.

Utilisation des informations d'authentification

Vous pouvez utiliser le widget d'informations d'authentification dans IBM MQ Console pour ajouter et supprimer des objets d'informations d'authentification sur un gestionnaire de files d'attente. Vous pouvez également afficher et définir les propriétés et gérer les enregistrements de droits d'accès pour les objets.


Avant de commencer

Vous devez créer un widget d'informations d'authentification avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir [«Utilisation des objets IBM MQ»](#), à la page 101.



Pourquoi et quand exécuter cette tâche


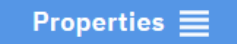

Le widget Informations d'authentification répertorie les informations d'authentification qui existent pour un gestionnaire de files d'attente spécifique. Vous pouvez sélectionner des informations d'authentification individuelles dans la liste à utiliser.

Les informations d'authentification du gestionnaire de files d'attente font partie du support IBM MQ pour le protocole TLS (Transport Layer Security). Ces objets contiennent les définitions requises pour effectuer la vérification de la révocation de certificat à l'aide d'OCSP ou de listes de révocation de certificat (CRL) sur les serveurs LDAP, ainsi que les définitions requises pour activer la vérification de l'ID utilisateur et du mot de passe.

 Vous ne pouvez pas utiliser IDPW LDAP, ni afficher ou éditer des enregistrements de droits d'accès pour un objet d'informations d'authentification sur z/OS.

Procédure

- Pour ajouter un objet d'informations d'authentification:
 - a) Cliquez sur l'icône de création  dans la barre d'outils du widget d'informations d'authentification.
 - b) Indiquez le nom de l'objet des informations d'authentification. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_" et "%".
 - c) Indiquez le type d'objet d'informations d'authentification.
 - d) Indiquez des informations supplémentaires appropriées pour le type d'objet:
 - Pour **CRL LDAP**, indiquez le **nom du serveur LDAP**. Il s'agit du nom d'hôte, de l'adresse IPv4 à notation décimale à point ou de la notation hexadécimale IPv6 de l'hôte sur lequel s'exécute le serveur LDAP, avec un numéro de port facultatif.
 - Pour **OCSP**, spécifiez l' **URL du répondeur OCSP**. Il s'agit de l'URL du programme de réponse utilisé pour vérifier la révocation de certificat. Il doit s'agir d'une adresse URL HTTP contenant le nom d'hôte et le numéro de port du canal répondeur OCSP. Si le canal répondeur OCSP utilise le port 80, qui est le port par défaut pour HTTP, le numéro de port peut être omis. Les adresses URL HTTP sont définies dans la norme RFC 1738.
 - Pour **IDPW OS**, aucune exigence supplémentaire n'est requise.
 - Pour **IDPW LDAP**, indiquez le **nom de serveur LDAP** et le nom d' **utilisateur abrégé** . Le nom du serveur LDAP est le nom d'hôte, IPv4 l'adresse décimale à point ou IPv6 la notation hexadécimale de l'hôte sur lequel s'exécute le serveur LDAP, avec un numéro de port facultatif. Le nom d'utilisateur abrégé est la zone de l'enregistrement utilisateur LDAP qui est utilisée comme nom abrégé pour la connexion.
 - e) Cliquez sur **Create**.
- Pour supprimer un objet d'informations d'authentification :
 - a) Dans le widget, sélectionnez l'objet d'informations d'authentification à supprimer de la liste.
 - b) Cliquez sur l'icône de suppression  dans la barre d'outils du widget.
 - c) Confirmez la suppression de l'objet d'informations d'authentification en cliquant sur **Supprimer**. L'objet est supprimé.
- Pour afficher et éditer les propriétés d'un objet d'informations d'authentification :
 - a) Sélectionnez l'objet d'informations d'authentification dans le widget.

- b) Cliquez sur l'icône Propriétés   dans la barre d'outils du widget. Vous pouvez également cliquer deux fois sur l'objet d'informations d'authentification.
- c) Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande.
- Pour afficher et éditer des enregistrements de droits d'accès pour un objet d'informations d'authentification:
 - a) Sélectionnez l'objet d'informations d'authentification dans le widget d'informations d'authentification.
 - b) Cliquez sur  ... > **Gérer les enregistrements de droits d'accès**. Les enregistrements de droits d'accès affichent les droits dont disposent les utilisateurs et les administrateurs sur l'objet d'informations d'authentification sélectionné.

Utilisation des abonnements

Vous pouvez utiliser le widget d'abonnements dans IBM MQ Console pour ajouter et supprimer des abonnements sur un gestionnaire de files d'attente, afficher et définir les propriétés, et gérer les enregistrements de droits d'accès pour les abonnements.

Avant de commencer

Vous devez créer un widget d'abonnements avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir [«Utilisation des objets IBM MQ»](#), à la page 101.



Pourquoi et quand exécuter cette tâche



Les abonnements sont envoyés à un gestionnaire de files d'attente et contiennent des informations sur les publications que l'abonné souhaite recevoir :

- Chaîne de rubrique qui intéresse l'abonné ; cette rubrique peut être résolue en plusieurs chaînes de rubrique si des caractères génériques sont utilisés.
- Une chaîne de sélection facultative à appliquer aux messages publiés.
- Nom de la file d'attente dans laquelle les publications sélectionnées sont placées.

Pour plus d'informations sur les abonnements, voir [Abonnés et abonnements](#) et [DEFINE SUB](#).

Procédure

- Pour ajouter un objet d'abonnement :
 - a) Cliquez sur l'icône Créer   dans la barre d'outils du widget Abonnements.
 - b) Spécifiez le nom de l'objet. Les caractères valides sont les lettres, les chiffres, ainsi que les caractères ".", "/", "_", et "%".
 - c) Dans la zone **Classe de destination**, sélectionnez **Gérée** ou **Fournie**. Si vous sélectionnez **Géré**, une destination est créée sur le gestionnaire de files d'attente local.
 - d) Si vous sélectionnez la classe de destination **Fournie**, dans la zone **Destination**, spécifiez le nom de la file d'attente vers laquelle les messages pour cet abonnement sont acheminés.
 - e) Dans la zone **Chaîne de rubrique**, spécifiez la chaîne de rubrique à laquelle vous voulez vous abonner.
 - f) Sélectionnez un paramètre **Utilisation des caractères génériques**. Sélectionnez **Caractère générique au niveau caractère** pour spécifier que les caractères génériques représentent des parties de chaîne. Sélectionnez **Caractère générique au niveau rubrique** pour spécifier que les caractères génériques représentent des parties de la hiérarchie de rubriques.

- g) Sélectionnez une **portée**. Sélectionnez **Tout** pour que l'abonnement soit acheminé vers tous les gestionnaires de files d'attente directement connectés via une hiérarchie ou une collectivité de publication/abonnement. Sélectionnez **Gestionnaire de files d'attente** pour que l'abonnement réachemine les messages qui sont publiés dans la rubrique uniquement dans ce gestionnaire de files d'attente.
- h) Facultatif : Spécifiez un **sélecteur**. Une chaîne de sélection est une expression appliquée à une publication afin de déterminer si elle correspond à un abonnement.
- i) Cliquez sur **Create**.
- Pour supprimer un objet d'abonnement :
 - a) Sélectionnez l'objet d'abonnement à supprimer dans la liste du widget d'abonnement.
 - b) Cliquez sur l'icône de suppression  dans la barre d'outils du widget.
 - c) Confirmez la suppression de l'objet d'abonnement en cliquant sur **Supprimer**. L'objet est supprimé.
- Pour afficher et éditer les propriétés d'un objet d'abonnement :
 - a) Sélectionnez l'objet d'abonnement dans le widget.
 - b) Cliquez sur l'icône Propriétés  dans la barre d'outils du widget. Vous pouvez également cliquer deux fois sur l'objet d'abonnement.
 - c) Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande.

Utilisation d'enregistrements d'authentification de canal

Vous pouvez utiliser le widget Enregistrements d'authentification de canal dans IBM MQ Console pour ajouter et supprimer des enregistrements d'authentification de canal sur un gestionnaire de files d'attente. Vous pouvez également afficher et définir les propriétés des enregistrements d'authentification de canal.

Avant de commencer

Vous devez créer un widget d'enregistrements d'authentification de canal avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir [«Utilisation des objets IBM MQ»](#), à la page 101.



Pourquoi et quand exécuter cette tâche

Pour exercer un contrôle plus précis sur l'accès accordé aux systèmes se connectant au niveau du canal, vous pouvez utiliser des enregistrements d'authentification de canal.

Pour appliquer la sécurité, vous pouvez utiliser des enregistrements d'authentification de canal bloquants afin de bloquer l'accès à vos canaux. Vous pouvez aussi utiliser des enregistrements d'authentification de canal de mappe d'adresses pour autoriser l'accès à des utilisateurs spécifiés. Pour en savoir plus sur les enregistrements d'authentification de canal, voir [Enregistrements d'authentification de canal](#).

Procédure

- Pour ajouter un enregistrement d'authentification de canal avec une identité de nom distinctif SSL/TLS, voir [«Création d'enregistrements d'authentification de canal avec une identité de nom distinctif SSL/TLS»](#), à la page 112.
- Pour ajouter un enregistrement d'authentification de canal avec une identité d'ID utilisateur d'application client, voir [«Création d'enregistrements d'authentification de canal avec une identité d'ID utilisateur d'application client»](#), à la page 113.

- Pour ajouter un enregistrement d'authentification de canal avec une identité de nom de gestionnaire de files d'attente éloignées, voir [«Création d'enregistrements d'authentification de canal avec une identité de nom de gestionnaire de files d'attente éloignées»](#), à la page 114.
- Pour ajouter un enregistrement d'authentification de canal avec une identité d'adresse, voir [«Création d'enregistrements d'authentification de canal avec une identité d'adresse IP»](#), à la page 115.
- Pour supprimer un enregistrement d'authentification de canal :
 - a) Sélectionnez l'enregistrement d'authentification de canal à supprimer dans la liste du widget Enregistrements d'authentification de canal.
 - b) Cliquez sur l'icône de suppression  dans la barre d'outils du widget.
 - c) Confirmez la suppression de l'enregistrement d'authentification de canal en cliquant sur **Supprimer**. L'enregistrement d'authentification de canal est supprimé.
- Pour afficher et éditer les propriétés d'un enregistrement d'authentification de canal:
 - a) Sélectionnez l'enregistrement d'authentification de canal à éditer dans la liste du widget d'enregistrement d'authentification de canal.
 - b) Cliquez sur l'icône Propriétés  dans la barre d'outils du widget. Vous pouvez également cliquer deux fois sur l'enregistrement d'authentification de canal.
 - c) Affichez les propriétés et éditez-les si nécessaire. Si la zone de texte de la propriété est désactivée, la propriété est en lecture seule ou peut être éditée uniquement à partir de la ligne de commande.


Création d'enregistrements d'authentification de canal avec une identité de nom distinctif SSL/TLS

Vous pouvez utiliser le widget Enregistrements d'authentification de canal pour créer des enregistrements d'authentification de canal d'autorisation, de blocage et d'avertissement avec une identité de nom distinctif SSL/TLS. L'identité du nom distinctif SSL/TLS correspond aux utilisateurs qui présentent un certificat personnel SSL ou TLS contenant un nom distinctif spécifié.

Avant de commencer

Vous devez créer un widget d'enregistrements d'authentification de canal avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir [«Utilisation des objets IBM MQ»](#), à la page 101.

Procédure

- Pour ajouter un enregistrement d'authentification de canal:
 - a) Cliquez sur l'icône de création  dans la barre d'outils du widget Enregistrement d'authentification de canal.
 - b) Sélectionnez le **Type de règle** pour indiquer le type de règle que vous souhaitez sur l'enregistrement d'authentification de canal:
 - Sélectionnez **Autoriser** pour autoriser l'accès aux connexions entrantes.
 - Sélectionnez **Bloquer** pour bloquer l'accès aux connexions entrantes.
 - Sélectionnez **Avertir** pour avertir de l'accès aux connexions entrantes qui seraient bloquées. L'accès à la connexion est autorisé et un message d'erreur est généré. Si des événements sont configurés, un message d'événement est créé pour afficher les détails de ce qui serait bloqué. Seules les règles correspondantes sont signalées.
 - c) Sélectionnez le type d'identité **Nom distinctif SSL/TLS** dans la liste.
 - d) Cliquez sur **Suivant**.
 - e) Indiquez un **profil de canal**.

Le profil de canal est le nom du canal ou de l'ensemble de canaux pour lesquels vous définissez l'authentification de canal. Le profil peut contenir des caractères génériques afin que vous puissiez bloquer une plage de canaux. Par exemple, le profil `alphadelta*` bloque les canaux nommés `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.

- f) Indiquez le **Nom de l'homologue**. Par exemple, `CN=John Smith, O=IBM, OU=Test, C=GB`. Pour plus d'informations sur les noms d'homologue, voir [WebSphere MQ pour les valeurs SSLPEER](#).
- g) Facultatif : Indiquez le filtre **Adresse** utilisé. L'adresse est l'adresse IP attendue à l'autre extrémité du canal.
- h) Facultatif : Indiquez le **nom de l'émetteur du certificat SSL**. Le nom de l'émetteur de certificat SSL est le nom de l'autorité de certification par laquelle le certificat SSL/TLS doit être émis.
- i) Facultatif : Cliquez sur **Suivant**.
- j) Facultatif : Pour un type de règle **Autoriser**, vous pouvez éventuellement spécifier la **source d'utilisateur** pour l'enregistrement d'authentification de canal. La source utilisateur indique la source de l'ID utilisateur utilisé lorsque la connexion entrante correspond au nom distinctif SSL/TLS.
 - L'option **Canal** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur de flux ou tout utilisateur défini sur l'objet de canal.
 - L'option **Mapper** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur spécifié dans la zone **ID utilisateur MCA**.
- k) Facultatif : Cliquez sur **Suivant**.
- l) Facultatif : Spécifiez une **Description** pour l'enregistrement d'authentification de canal.
- m) Cliquez sur **Create**. Le nouvel enregistrement d'authentification de canal est créé.

V 9.0.1 *Création d'enregistrements d'authentification de canal avec une identité d'ID utilisateur d'application client*


Vous pouvez utiliser le widget Enregistrements d'authentification de canal pour créer des enregistrements d'authentification de canal d'autorisation, de blocage et d'avertissement avec une identité d'ID utilisateur d'application client. L'identité de l'ID utilisateur de l'application client correspond aux ID d'application client d'un canal de connexion client.

Avant de commencer

Vous devez créer un widget d'enregistrements d'authentification de canal avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ, voir «[Utilisation des objets IBM MQ](#)», à la page 101.

Procédure

- Pour ajouter un enregistrement d'authentification de canal:

- a) Cliquez sur l'icône de création  dans la barre d'outils du widget Enregistrement d'authentification de canal.
- b) Sélectionnez le **Type de règle** pour indiquer le type de règle que vous souhaitez sur l'enregistrement d'authentification de canal:
 - Sélectionnez **Autoriser** pour autoriser l'accès aux connexions entrantes.
 - Sélectionnez **Bloquer** pour bloquer l'accès aux connexions entrantes.
 - Sélectionnez **Avertir** pour avertir de l'accès aux connexions entrantes qui seraient bloquées. L'accès à la connexion est autorisé et un message d'erreur est généré. Si des événements sont configurés, un message d'événement est créé pour afficher les détails de ce qui serait bloqué. Seules les règles correspondantes sont signalées.
- c) Sélectionnez le type d'identité **ID utilisateur de l'application client** dans la liste.
- d) Cliquez sur **Suivant**.

- e) Indiquez un **profil de canal**.
Le nom de profil de canal est le nom du canal ou de l'ensemble de canaux pour lequel vous définissez l'authentification de canal. Le profil peut contenir des caractères génériques afin que vous puissiez bloquer une plage de canaux. Par exemple, le profil `a1phade1ta*` bloque les canaux nommés `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
- f) Indiquez l' **ID utilisateur client**. L'ID utilisateur client est l'ID utilisateur du client que vous souhaitez autoriser, bloquer ou avertir.
- g) Facultatif : Indiquez le filtre **Adresse** utilisé. L'adresse est l'adresse IP attendue à l'autre extrémité du canal.
- h) Facultatif : Cliquez sur **Suivant**.
- i) Facultatif : Pour un type de règle **Autoriser** , vous pouvez éventuellement spécifier la **source d'utilisateur** pour l'enregistrement d'authentification de canal. La source utilisateur indique la source de l'ID utilisateur utilisé lorsque la connexion entrante correspond à l'ID utilisateur client.
 - L'option **Canal** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur de flux ou tout utilisateur défini sur l'objet de canal.
 - L'option **Mapper** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur spécifié dans la zone **ID utilisateur MCA** .
- j) Facultatif : Cliquez sur **Suivant**.
- k) Facultatif : Spécifiez une **Description** pour l'enregistrement d'authentification de canal.
- l) Cliquez sur **Create**. Le nouvel enregistrement d'authentification de canal est créé.


V 9.0.1 *Création d'enregistrements d'authentification de canal avec une identité de nom de gestionnaire de files d'attente éloignées*

Vous pouvez utiliser le widget Enregistrements d'authentification de canal pour créer des enregistrements d'authentification de canal d'autorisation, de blocage et d'avertissement avec une identité de nom de gestionnaire de files d'attente éloignées. L'identité du nom de gestionnaire de files d'attente éloignées correspond au gestionnaire de files d'attente spécifié.

Avant de commencer

Vous devez créer un widget d'enregistrements d'authentification de canal avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ , voir [«Utilisation des objets IBM MQ»](#), à la page 101.

Procédure

- Pour ajouter un enregistrement d'authentification de canal:
 - a) Cliquez sur l'icône de création  dans la barre d'outils du widget Enregistrement d'authentification de canal.
 - b) Sélectionnez le **Type de règle** pour indiquer le type de règle que vous souhaitez sur l'enregistrement d'authentification de canal:
 - Sélectionnez **Autoriser** pour autoriser l'accès aux connexions entrantes.
 - Sélectionnez **Bloquer** pour bloquer l'accès aux connexions entrantes.
 - Sélectionnez **Avertir** pour avertir de l'accès aux connexions entrantes qui seraient bloquées. L'accès à la connexion est autorisé et un message d'erreur est généré. Si des événements sont configurés, un message d'événement est créé pour afficher les détails de ce qui serait bloqué. Seules les règles correspondantes sont signalées.
 - c) Sélectionnez le type d'identité **Nom du gestionnaire de files d'attente éloignées** dans la liste.
 - d) Cliquez sur **Suivant**.
 - e) Indiquez un **nom de profil**.

Le nom de profil est le nom du canal ou de l'ensemble de canaux pour lequel vous définissez l'authentification de canal. Le profil peut contenir des caractères génériques afin que vous puissiez bloquer une plage de canaux. Par exemple, le profil `alphadelta*` bloque les canaux nommés `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.

- f) Indiquez le **nom du gestionnaire de files d'attente**. Le nom du gestionnaire de files d'attente indique le nom du gestionnaire de files d'attente éloignées que vous souhaitez autoriser, bloquer ou avertir.
- g) Facultatif : Indiquez le filtre **Adresse** utilisé. L'adresse est l'adresse IP attendue à l'autre extrémité du canal.
- h) Facultatif : Cliquez sur **Suivant**.
- i) Facultatif : Pour un type de règle **Autoriser** , vous pouvez éventuellement spécifier la **source d'utilisateur** pour l'enregistrement d'authentification de canal. La source utilisateur indique la source de l'ID utilisateur utilisé lorsque la connexion entrante correspond au nom du gestionnaire de files d'attente éloignées.
 - L'option **Canal** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur de flux ou tout utilisateur défini sur l'objet de canal.
 - L'option **Mapper** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur spécifié dans la zone **ID utilisateur MCA** .
- j) Facultatif : Cliquez sur **Suivant**.
- k) Facultatif : Spécifiez une **Description** pour l'enregistrement d'authentification de canal.
- l) Cliquez sur **Create**. Le nouvel enregistrement d'authentification de canal est créé.

V 9.0.1 *Création d'enregistrements d'authentification de canal avec une identité d'adresse IP*


Vous pouvez utiliser le widget Enregistrements d'authentification de canal pour créer des enregistrements d'authentification de canal d'autorisation, de blocage et d'avertissement avec une identité d'adresse. L'identité de l'adresse correspond à des adresses IP spécifiques.

Avant de commencer

Vous devez créer un widget d'enregistrements d'authentification de canal avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ , voir [«Utilisation des objets IBM MQ»](#), à la page 101.

Procédure

- Pour ajouter un enregistrement d'authentification de canal:

- a) Cliquez sur l'icône de création  dans la barre d'outils du widget Enregistrement d'authentification de canal.
- b) Sélectionnez le **Type de règle** pour indiquer le type de règle que vous souhaitez sur l'enregistrement d'authentification de canal:
 - Sélectionnez **Autoriser** pour autoriser l'accès aux connexions entrantes.
 - Sélectionnez **Bloquer** pour bloquer l'accès aux connexions entrantes.
 - Sélectionnez **Avertir** pour avertir de l'accès aux connexions entrantes qui seraient bloquées. L'accès à la connexion est autorisé et un message d'erreur est généré. Si des événements sont configurés, un message d'événement est créé pour afficher les détails de ce qui serait bloqué. Seules les règles correspondantes sont signalées.
- c) Sélectionnez le type d'identité **Adresse** dans la liste.
- d) Cliquez sur **Suivant**.
- e) Facultatif : Pour un type de règle **Block** ou **Warn** , spécifiez **When to match**.

Vous pouvez choisir parmi les options suivantes:

- **Au niveau du programme d'écoute.** Cette option tente de faire correspondre la règle au niveau du programme d'écoute.
 - **Sur le canal.** Cette option tente de faire correspondre la règle au niveau du canal.
- f) Indiquez un **nom de profil**.
Le nom de profil est le nom du canal ou de l'ensemble de canaux pour lequel vous définissez l'authentification de canal. Le profil peut contenir des caractères génériques afin que vous puissiez bloquer une plage de canaux. Par exemple, le profil `alphaDelta*` bloque les canaux nommés `alphaDelta1`, `alphaDelta2`, `alphaDelta3`, etc.
- g) Indiquez une **adresse**. L'adresse est l'adresse IP ou une liste d'adresses IP séparées par des virgules qui sont autorisées ou bloquées.
- h) Facultatif : Cliquez sur **Suivant**.
- i) Facultatif : Pour un type de règle **Autoriser** , vous pouvez éventuellement spécifier la **source d'utilisateur** pour l'enregistrement d'authentification de canal. La source utilisateur indique la source de l'ID utilisateur utilisé lorsque la connexion entrante correspond au nom du gestionnaire de files d'attente éloignées.
- L'option **Canal** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur de flux ou tout utilisateur défini sur l'objet de canal.
 - L'option **Mapper** indique que les connexions entrantes qui correspondent au mappage utilisent l'ID utilisateur spécifié dans la zone **ID utilisateur MCA** .
- j) Facultatif : Cliquez sur **Suivant**.
- k) Facultatif : Spécifiez une **Description** pour l'enregistrement d'authentification de canal.
- l) Cliquez sur **Create**. Le nouvel enregistrement d'authentification de canal est créé.


V 9.0.1 *Création d'enregistrements d'authentification de canal avec une identité d'ID utilisateur affectée finale*

Vous pouvez utiliser le widget Enregistrements d'authentification de canal pour créer des enregistrements d'authentification de canal de blocage et d'avertissement avec une identité d'ID utilisateur affectée finale. L'identité de l'ID utilisateur final correspond à la liste des ID utilisateur spécifiés à partir d'un canal serveur.

Avant de commencer

Vous devez créer un widget d'enregistrements d'authentification de canal avant de pouvoir l'utiliser. Pour plus d'informations sur la création de widgets d'objet IBM MQ , voir [«Utilisation des objets IBM MQ»](#), à la page 101.

Procédure

- Pour ajouter un enregistrement d'authentification de canal:
 - a) Cliquez sur l'icône de création **V 9.0.5** **Create**  dans la barre d'outils du widget Enregistrement d'authentification de canal.
 - b) Sélectionnez le **Type de règle** pour indiquer le type de règle que vous souhaitez sur l'enregistrement d'authentification de canal:
 - Sélectionnez **Bloquer** pour bloquer l'accès aux connexions entrantes.
 - Sélectionnez **Avertir** pour avertir de l'accès aux connexions entrantes qui seraient bloquées. L'accès à la connexion est autorisé et un message d'erreur est généré. Si des événements sont configurés, un message d'événement est créé pour afficher les détails de ce qui serait bloqué. Seules les règles correspondantes sont signalées.
 - c) Sélectionnez le type d'identité **ID utilisateur final affecté** dans la liste.

- d) Cliquez sur **Suivant**.
- e) Indiquez un **nom de profil**.
Le nom de profil est le nom du canal ou de l'ensemble de canaux pour lequel vous définissez l'authentification de canal. Le profil peut contenir des caractères génériques afin que vous puissiez bloquer une plage de canaux. Par exemple, le profil `alphadelta*` bloque les canaux nommés `alphadelta1`, `alphadelta2`, `alphadelta3`, etc.
- f) Indiquez la **liste d'utilisateurs**. La liste d'utilisateurs est une liste d'ID utilisateur séparés par des virgules à bloquer à partir du canal.
- g) Facultatif : Cliquez sur **Suivant**.
- h) Facultatif : Spécifiez une **Description** pour l'enregistrement d'authentification de canal.
- i) Cliquez sur **Create**. Le nouvel enregistrement d'authentification de canal est créé.

V 9.0.1

Utilisation d'enregistrements de droits d'accès


Vous pouvez contrôler l'accès dont les groupes disposent pour les gestionnaires de files d'attente et les objets IBM MQ en spécifiant un enregistrement de droits d'accès pour ces groupes.

Pourquoi et quand exécuter cette tâche

Vous pouvez optimiser l'accès d'un groupe d'utilisateurs de messagerie à un gestionnaire de files d'attente ou à un objet IBM MQ particulier à l'aide d'enregistrements de droits d'accès. Vous configurez l'enregistrement de droits d'accès de la même façon pour tous les types d'objet en suivant la même procédure, bien que les droits d'accès réels que vous configurez dépendent du type d'objet.

Par exemple, comparez les différents droits disponibles pour un gestionnaire de files d'attente et une file d'attente, comme illustré dans les images suivantes: V 9.0.5

Authority records for 'qm3'

Delete  1 item selected [Cancel](#)

| ▲ Entity name | Entity type |
|---------------|-------------|
| mqm | Group |
| mqsystem | User |

Total: 2 Last updated: 3:40:18 PM

Administration

- Change
- Delete
- Display
- Ctrl

Context

- Set all context
- Set identity context

MQI

- Alternate user authority
- Connect
- Inquire
- Set
- System

Check all


Uncheck all

Close

Save

V 9.0.5

Authority records for 'q1' on qm3

| Delete  | | 1 item selected | Cancel |
|--|-------------|-----------------|--------|
| ▲ Entity name | Entity type | | |
| mqm | Group | | |
| mqsystem | User | | |

| Administration | Context | MQI |
|---|---|---|
| <input checked="" type="checkbox"/> Change | <input checked="" type="checkbox"/> Pass all context | <input checked="" type="checkbox"/> Browse |
| <input checked="" type="checkbox"/> Clear | <input checked="" type="checkbox"/> Pass identity context | <input checked="" type="checkbox"/> Inquire |
| <input checked="" type="checkbox"/> Delete | <input checked="" type="checkbox"/> Set all context | <input checked="" type="checkbox"/> Get |
| <input checked="" type="checkbox"/> Display | <input checked="" type="checkbox"/> Set identity context | <input checked="" type="checkbox"/> Put |
| | | <input checked="" type="checkbox"/> Set |


Buttons:



Vous ne pouvez pas utiliser les enregistrements de droits d'accès sur z/OS.

Procédure

- Pour afficher ou éditer un enregistrement de droits d'accès pour un objet IBM MQ :
 - a) Sélectionnez l'objet dans un widget dans le tableau de bord. Le gestionnaire de files d'attente associé doit être en cours d'exécution.
 - b) Dans la barre d'outils du widget approprié, sélectionnez ... > **Gérer les enregistrements de droits d'accès**.
 - c) Sélectionnez le groupe pour lequel afficher l'enregistrement de droits d'accès. Les droits d'accès pour ce groupe sont affichés.
 - d) Sélectionnez ou effacez les droits requis. Des droits d'accès différents sont disponibles en fonction du type d'objet pour lequel vous créez un enregistrement de droits d'accès.
 - e) Cliquer sur **Sauvegarder**.
- Afin d'afficher ou d'éditer un enregistrement de droits de création pour un gestionnaire de files d'attente :
 - a) Sélectionnez le gestionnaire de files d'attente dans un widget de gestionnaire de files d'attente sur le tableau de bord. Il doit être en cours d'exécution.
 - b) Dans la barre d'outils du widget, sélectionnez ... > **Gérer les enregistrements de création de droits d'accès**.
 - c) Sélectionnez le groupe pour lequel afficher l'enregistrement de droits de création. Les droits d'accès pour ce groupe sont affichés.

- d) Sélectionnez ou désélectionnez les droits de création selon les besoins.
- e) Cliquer sur **Sauvegarder**.
- Pour créer un enregistrement de droits d'accès pour un objet IBM MQ :
 - a) Sélectionnez l'objet IBM MQ dans un widget du tableau de bord. Le gestionnaire de files d'attente associé doit être en cours d'exécution.
 - b) Dans la barre d'outils du widget, sélectionnez ... > **Gérer les enregistrements de droits d'accès**.
 - c) Cliquez sur l'icône plus .
 - d) Indiquez le nom de l'utilisateur ou du groupe pour lequel vous créez l'enregistrement de droits d'accès. L'utilisateur ou le groupe doit exister.
 - e) Sélectionnez **Type d'entité** pour indiquer si l'entité est un utilisateur ou un groupe.
 - f) Cliquez sur **Create**.
 - g) Sélectionnez ou désélectionnez les droits que l'utilisateur ou le groupe doit posséder. Des droits d'accès différents sont disponibles pour chaque type d'objet.
 - h) Cliquer sur **Sauvegarder**.
- Afin de créer un enregistrement de droits d'accès pour créer des objets sur un gestionnaire de files d'attente :
 - a) Sélectionnez le gestionnaire de files d'attente dans un widget dans le tableau de bord. Il doit être en cours d'exécution.
 - b) Dans la barre d'outils du widget, sélectionnez ... > **Gérer les enregistrements de création de droits d'accès**.
 - c) Cliquez sur l'icône Créer .
 - d) Indiquez le nom de l'utilisateur ou du groupe pour lequel vous créez l'enregistrement de droits d'accès. L'utilisateur ou le groupe doit exister.
 - e) Sélectionnez **Type d'entité** pour indiquer si l'entité est un utilisateur ou un groupe.
 - f) Cliquez sur **Create**.
 - g) Sélectionnez ou désélectionnez les droits de création que vous souhaitez donner à l'utilisateur ou au groupe.
 - h) Cliquer sur **Sauvegarder**.

Surveillance de l'utilisation des ressources système

Le widget Graphiques du IBM MQ Console permet d'afficher les données de surveillance des gestionnaires de files d'attente.

Pourquoi et quand exécuter cette tâche

Vous ajoutez un widget Graphiques à votre tableau de bord, puis vous le configurez pour surveiller un aspect particulier de l'utilisation des ressources. Vous pouvez créer de nombreuses instances du widget Graphiques pour afficher différentes données. Les données sont affichées dans un format graphique.

Les données sont collectées à des intervalles de 10 secondes. L'axe des X du graphique affiche une chronologie. L'axe des Y affiche les unités appropriées à la ressource que vous affichez. Il est redimensionné dynamiquement pour adapter les données renvoyées.

Un gestionnaire de files d'attente au moins doit être en cours d'exécution pour que vous puissiez configurer un widget de graphique.

 Vous ne pouvez pas surveiller l'utilisation des ressources système sous z/OS.

Procédure

1. Ajoutez un widget Graphiques à votre tableau de bord:

a) Cliquez sur l'icône Ajouter un widget

V 9.0.5

Add widget

b) Sélectionnez **Graphiques**.

2. Configurez le widget Graphiques pour afficher les données:

a) Cliquez sur l'icône de configuration

V 9.0.5



dans la barre de titre du widget Graphiques.

b) Facultatif : Entrez un **titre de widget**. Ce titre est affiché dans la barre de titre du widget.

c) Sélectionnez la **classe de ressources** à surveiller:

Unités centrales de plateforme

Surveillez l'utilisation des UC.

Magasins de données persistantes de plateforme

Surveillez l'utilisation de la ressource de disque.

Statistiques d'utilisation d'API

Surveillez les appels d'API.

Statistiques d'utilisation d'API par file d'attente

Surveillance des appels d'API par des files d'attente individuelles. Lorsque vous choisissez cette classe, vous indiquez le nom de la file d'attente à surveiller dans la zone **Objet**.

d) Sélectionnez le **Type de ressource** à surveiller.

Les types de ressource disponibles pour la sélection dépendent de la classe de ressource sélectionnée. Le tableau suivant présente les types de ressource:

| Classe | Tapez | Description |
|--|---|--|
| Unités centrales de plateforme | Performance d'UC - niveau plateforme | Sélectionnez ce type pour afficher les données de performances des unités centrales et de la mémoire. |
| | Performance d'UC - gestionnaire de files d'attente en cours d'exécution | Sélectionnez ce type pour afficher les données de performance des unités centrales et de la mémoire qui sont liées aux gestionnaires de files d'attente que vous surveillez. Un gestionnaire de files d'attente doit être en cours d'exécution pour que vous puissiez le surveiller. Si vous surveillez les résultats pour plusieurs gestionnaires de files d'attente, des couleurs différentes sont utilisées pour distinguer les données de performance dans le graphique. |
| Magasins de données persistantes de plateforme | Utilisation de disque - niveau plateforme | Sélectionnez ce type pour afficher les données de performance relatives à l'utilisation globale du disque. |

| <i>Tableau 6. Types de ressource (suite)</i> | | |
|---|---|---|
| Classe | Tapez | Description |
| | Utilisation de disque - gestionnaires de files d'attente en cours d'exécution | Sélectionnez ce type afin d'afficher les données de performance pour l'utilisation du disque liée aux gestionnaires de files d'attente que vous surveillez. Un gestionnaire de files d'attente doit être en cours d'exécution pour que vous puissiez le surveiller. Si vous surveillez les résultats pour plusieurs gestionnaires de files d'attente, des couleurs différentes sont utilisées pour distinguer les données de performance dans le graphique. |
| | Utilisation de disque - journal de reprise du gestionnaire de files d'attente | Sélectionnez ce type pour afficher les données sur la façon dont le stockage sur disque est utilisé pour le journal de reprise de chaque gestionnaire de files d'attente que vous surveillez. |
| Statistiques d'utilisation d'API | MQCONN et MQDISC | Sélectionnez ce type pour afficher des données sur les appels MQCONN et MQDISC. |
| | MQOPEN et MQCLOSE | Sélectionnez ce type pour afficher des données sur les appels MQOPEN et MQCLOSE. |
| | MQINQ et MQSET | Sélectionnez ce type pour afficher des données sur les appels MQINQ et MQSET. |
| | MQPUT | Sélectionnez ce type pour afficher des données sur les appels liés à MQPUT. |
| | MQGET | Sélectionnez ce type pour afficher des données sur les appels liés à MQGET. |
| | Validation et annulation | Sélectionnez ce type pour afficher des informations sur l'utilisation des points de synchronisation par le gestionnaire de files d'attente. |
| | S'abonner | Sélectionnez ce type pour afficher les données relatives aux appels MQSUB. |
| | Publication | Sélectionnez ce type pour afficher des données sur les messages publiés. |
| Statistiques d'utilisation d'API par file d'attente | MQOPEN et MQCLOSE | Sélectionnez ce type afin d'afficher des données sur les appels MQOPEN et MQCLOSE pour la file d'attente spécifiée. |
| | MQINQ et MQSET | Sélectionnez ce type afin d'afficher des données sur les appels MQINQ et MQSET pour la file d'attente spécifiée. |

| <i>Tableau 6. Types de ressource (suite)</i> | | |
|--|-----------------|--|
| Classe | Tapez | Description |
| | MQPUT et MQPUT1 | Sélectionnez ce type afin d'afficher des données sur les appels liés à MQPUT et à MQPUT1 pour la file d'attente spécifiée. |
| | MQGET | Sélectionnez ce type afin d'afficher des données sur les appels liés à MQGET pour la file d'attente spécifiée. |

e) Sélectionnez l' **élément de ressource** à surveiller:

Les éléments de ressource pouvant être sélectionnés dépendent de la classe de ressource et du type de ressource sélectionnés. Les tableaux suivants présentent les éléments de ressource:

| <i>Tableau 7. Eléments pour les ressources des unités centrales de traitement de la plateforme</i> | | |
|--|--|---|
| Tapez | Élément | Description |
| Performance d'UC - niveau plateforme | Pourcentage de temps UC utilisateur | Affiche le pourcentage d'unité centrale occupée à l'état utilisateur. |
| | Pourcentage de temps UC système | Affiche le pourcentage d'unité centrale occupée à l'état système. |
| | Charge de l'unité centrale-moyenne sur une minute | Affiche la charge moyenne sur une minute. |
| | Charge de l'UC-moyenne sur cinq minutes | Affiche la charge moyenne sur 5 minutes. |
| | Charge de l'UC-moyenne sur quinze minutes | Affiche la charge moyenne sur quinze minutes. |
| | Pourcentage de mémoire vive disponible | Affiche le pourcentage de mémoire vive disponible. |
| | Nombre total d'octets de mémoire vive | Affiche le nombre total d'octets de mémoire vive configuré. |
| Performance d'UC - gestionnaire de files d'attente en cours d'exécution | Temps UC utilisateur - estimation du pourcentage pour le gestionnaire de files d'attente | Estime le pourcentage d'utilisation de l'unité centrale à l'état utilisateur pour les processus liés aux gestionnaires de files d'attente surveillés. |
| | Temps UC système - estimation du pourcentage pour le gestionnaire de files d'attente | Estime le pourcentage d'utilisation de l'UC à l'état système pour les processus liés aux gestionnaires de files d'attente surveillés. |
| | RAM total bytes - estimate for queue managers | Estime le nombre total d'octets de mémoire vive utilisé par les gestionnaires de files d'attente qui sont surveillés. |

| <i>Tableau 8. Eléments pour les ressources de stockage de persistance des données de la plateforme</i> | | |
|--|--|---|
| Tapez | Elément | Description |
| Utilisation de disque - niveau plateforme | Système de fichiers de trace MQ - octets utilisés | Affiche le nombre d'octets de stockage sur disque utilisés par le système de fichiers de trace. |
| | Système de fichiers de trace MQ - espace disponible | Affiche le stockage sur disque qui est réservé au système de fichiers de trace et qui est libre. |
| | Système de fichiers d'erreurs MQ - octets utilisés | Affiche le nombre d'octets de stockage sur disque utilisés par les données d'erreur. |
| | Système de fichiers d'erreurs MQ - espace disponible | Affiche le stockage sur disque réservé aux données d'erreur libres. |
| | Nombre de fichiers FDC MQ | Affiche le nombre en cours de fichiers FDC. |
| Utilisation de disque - gestionnaires de files d'attente en cours d'exécution | Système de fichiers du gestionnaire de files d'attente - octets utilisés | Affiche le nombre d'octets de stockage sur disque utilisés par les fichiers de gestionnaire de files d'attente pour les gestionnaires de files d'attente que vous surveillez. |
| | Système de fichiers du gestionnaire de files d'attente - espace disponible | Affiche l'espace de stockage sur disque réservé aux fichiers de gestionnaire de files d'attente libres. |
| Utilisation de disque - journal de reprise du gestionnaire de files d'attente | Journal - octets utilisés | Affiche le nombre d'octets de stockage sur disque utilisés pour les journaux de reprise des gestionnaires de files d'attente que vous surveillez. |
| | Journal - nombre d'octets maximal | Affiche le nombre maximal d'octets de stockage sur disque configurés pour être utilisés pour les journaux de reprise du gestionnaire de files d'attente. |
| | Système de fichier journal - octets utilisés | Affiche le nombre total d'octets du disque utilisés pour le système de fichier journal. |
| | Système de fichier journal - nombre d'octets maximal | Affiche le nombre d'octets du disque configurés pour le système de fichier journal. |
| | Journal - octets physiques écrits | Affiche le nombre d'octets physiques écrits dans les journaux de reprise. |
| | Journal - octets logiques écrits | Affiche le nombre d'octets logiques écrits dans les journaux de reprise. |

Tableau 8. Eléments pour les ressources de stockage de persistance des données de la plateforme (suite)

| Tapez | Elément | Description |
|--------------|--------------------------------------|---|
| | Journal - temps d'attente d'écriture | Affiche une mesure de temps d'attente lors de l'écriture synchrone dans le journal de reprise d'un gestionnaire de files d'attente. |

Tableau 9. Eléments pour les ressources de statistiques d'utilisation d'API

| Tapez | Elément | Description |
|-------------------|---|---|
| MQCONN et MQDISC | Nombre de MQCONN/MQCONNX | Affiche le nombre d'appels de MQCONN et MQCONNX. |
| | Echec du calcul du nombre de MQCONN/MQCONNX | Affiche le nombre d'appels de MQCONN et MQCONNX ayant échoué. |
| | Connexions simultanées - cote d'alerte haute | Affiche le nombre maximal de connexions simultanées dans l'intervalle de statistiques en cours. |
| | Nombre de MQDISC | Affiche le nombre d'appels de MQDISC. |
| MQOPEN et MQCLOSE | Nombre de MQOPEN | Affiche le nombre d'appels de MQOPEN. |
| | Echec du calcul du nombre de MQOPEN | Affiche le nombre d'appels de MQOPEN ayant échoué. |
| | Nombre de MQCLOSE | Affiche le nombre d'appels de MQCLOSE. |
| | Echec du calcul du nombre de MQCLOSE | Affiche le nombre d'appels de MQCLOSE ayant échoué. |
| MQINQ et MQSET | Nombre de MQINQ | Affiche le nombre d'appels de MQINQ. |
| | Echec du calcul du nombre de MQINQ | Affiche le nombre d'appels de MQINQ ayant échoué. |
| | Nombre de MQSET | Affiche le nombre d'appels de MQSET. |
| | Echec du calcul du nombre de MQSET | Affiche le nombre d'appels de MQSET ayant échoué. |
| MQPUT | Nombre de MQPUT/MQPUT1 d'intervalle total | Affiche le nombre d'appels de MQPUT et MQPUT1. |
| | Nombre d'octets MQPUT/MQPUT1 d'intervalle total | Affiche le nombre total d'octets de données insérés par des appels de MQPUT et MQPUT1. |

Tableau 9. Eléments pour les ressources de statistiques d'utilisation d'API (suite)

| Tapez | Elément | Description |
|--------------|---|---|
| | Nombre de MQPUT de message non persistant | Affiche le nombre de messages non persistants insérés par MQPUT. |
| | Nombre de MQPUT de message persistant | Affiche le nombre de messages persistants insérés par MQPUT. |
| | Echec du calcul du nombre de MQPUT | Affiche le nombre d'appels de MQPUT ayant échoué. |
| | Nombre de MQPUT1 de message non persistant | Affiche le nombre de messages non persistants insérés par MQPUT1. |
| | Nombre de MQPUT1 de message persistant | Affiche le nombre de messages persistants insérés par MQPUT1. |
| | Echec du calcul du nombre de MQPUT1 | Affiche le nombre d'appels de MQPUT1 ayant échoué. |
| | Put non-persistent message - byte count | Affiche le nombre d'octets insérés dans des messages non persistants. |
| | Put persistant message - byte count | Affiche le nombre d'octets insérés dans des messages persistants. |
| | Nombre de MQSTAT | Affiche le nombre d'appels de MQSTAT. |
| | Echec du calcul du nombre de MQSTAT | Affiche le nombre d'appels de MQSTAT ayant échoué. |
| MQGET | Interval total destructive get - count | Nombre de messages supprimés des files d'attente par MQGET. |
| | Commande get destructive d'intervalle total - nombre d'octets | Octets de données supprimés des files d'attente par MQGET. |
| | Commande get destructive de message non persistant - nombre | Nombre de messages non persistants supprimés des files d'attente par MQGET. |
| | Commande get destructive de message persistant - nombre | Nombre de messages persistants supprimés des files d'attente par MQGET. |
| | Echec de MQGET - nombre | Affiche le nombre d'appels de MQGET ayant échoué. |

| <i>Tableau 9. Eléments pour les ressources de statistiques d'utilisation d'API (suite)</i> | | |
|--|---|--|
| Tapez | Elément | Description |
| | Messages non persistants obtenus - nombre d'octets | Affiche le nombre d'octets de messages non persistants renvoyés à MQGET. |
| | Messages persistants obtenus - nombre d'octets | Affiche le nombre d'octets de messages persistants renvoyés à MQGET. |
| | Visualisation de messages non persistants - nombre | Affiche le nombre de messages non persistants visualisés. |
| | Visualisation de messages persistants - nombre | Affiche le nombre de messages persistants visualisés. |
| | Echec du calcul du nombre de visualisation | Affiche le nombre de visualisations de message ayant échoué. |
| | Visualisation de messages non persistants - nombre d'octets | Affiche le nombre d'octets de messages non persistants visualisés. |
| | Visualisation de messages persistants - nombre d'octets | Affiche le nombre d'octets de messages persistants visualisés. |
| | Nombre de messages arrivés à expiration | Affiche le nombre de messages arrivés à expiration. |
| | Nombre de files d'attente purgées | Affiche le nombre de files d'attente qui ont été purgées. |
| | Nombre de MQCB | Affiche le nombre d'appels de MQCB. |
| | Echec du calcul du nombre de MQCB | Affiche le nombre d'appels de MQCB ayant échoué. |
| | Nombre de MQCTL | Affiche le nombre d'appels de MQCTL. |
| | Echec du calcul du nombre de MQCTL | Affiche le nombre d'appels de MQCTL ayant échoué. |
| Validation et annulation | Nombre de validations | Affiche le nombre d'appels de MQCMIT. |
| | Echec du calcul du nombre de validations | Affiche le nombre d'appels de MQCMIT ayant échoué. |
| | Nombre d'annulations | Affiche le nombre d'appels de MQBACK. |

| <i>Tableau 9. Eléments pour les ressources de statistiques d'utilisation d'API (suite)</i> | | |
|--|--|---|
| Tapez | Elément | Description |
| S'abonner | Création du nombre d'abonnements durables | Affiche le nombre d'appels de MQSUB pour créer des abonnements durables. |
| | Modification du nombre d'abonnements durables | Affiche le nombre d'appels de MQSUB pour modifier des abonnements durables. |
| | Reprise du nombre d'abonnements durables | Affiche le nombre d'appels de MQSUB pour reprendre des abonnements durables. |
| | Création du nombre d'abonnements non durables | Affiche le nombre d'appels de MQSUB pour créer des abonnements non durables. |
| | Modification du nombre d'abonnements non durables | Affiche le nombre d'appels de MQSUB pour modifier des abonnements non durables. |
| | Reprise du nombre d'abonnements non durables | Affiche le nombre d'appels de MQSUB pour reprendre des abonnements non durables. |
| | Echec de la création/ modification/reprise du calcul du nombre d'abonnements | Affiche le nombre d'appels de MQSUBRQ ayant échoué pour créer, modifier ou reprendre des abonnements. |
| | Suppression du nombre d'abonnements durables | Affiche le nombre d'appels de MQSUB pour supprimer des abonnements durables. |
| | Suppression du nombre d'abonnements non durables | Affiche le nombre d'appels de MQSUB pour supprimer des abonnements non durables. |
| | Nombre d'échec de suppression d'abonnement | Affiche le nombre d'appels de MQSUB pour supprimer des abonnements. |
| | Nombre de MQSUBRQ | Affiche le nombre d'appels de MQSUBRQ. |
| | Echec du calcul du nombre de MQSUBRQ | Affiche le nombre d'appels de MQSUBRQ ayant échoué. |
| | Abonné durable - cote d'alerte haute | Affiche le nombre maximal d'abonnements durables dans l'intervalle de statistiques en cours. |
| | Abonné durable - cote d'alerte basse | Affiche le nombre minimal d'abonnements durables dans l'intervalle de statistiques en cours. |

| <i>Tableau 9. Eléments pour les ressources de statistiques d'utilisation d'API (suite)</i> | | |
|--|---|--|
| Tapez | Elément | Description |
| | Abonné non durable - cote d'alerte haute | Affiche le nombre maximal d'abonnements non durables dans l'intervalle de statistiques en cours. |
| | Abonné non durable - cote d'alerte basse | Affiche le nombre minimal d'abonnements non durables dans l'intervalle de statistiques en cours. |
| Publication | Intervalle total MQPUT/MQPUT1 de rubrique | Nombre de messages insérés dans les rubriques. |
| | Octets de rubrique d'intervalle total insérés | Nombre d'octets de message placés dans des rubriques. |
| | Publication aux abonnés - nombre de messages | Affiche le nombre de messages publiés pour les abonnés. |
| | Publication aux abonnés - nombre d'octets | Affiche le nombre d'octets des messages publiés pour les abonnés. |
| | Non persistant - nombre de MQPUT/MQPUT1 de rubrique | Affiche le nombre de messages non persistants insérés dans les rubriques. |
| | Persistant - nombre de MQPUT/MQPUT1 de rubrique | Affiche le nombre de messages persistants insérés dans les rubriques. |
| | Echec du calcul du nombre de MQPUT/MQPUT1 de rubrique | Affiche le nombre d'échecs de placement dans une rubrique. |

| <i>Tableau 10. Eléments pour les ressources de statistiques d'utilisation de l'API par file d'attente</i> | | |
|---|------------------------|--|
| Tapez | Elément | Description |
| MQOPEN et MQCLOSE | Nombre de MQOPEN | Affiche le nombre d'appels de MQOPEN. |
| | Nombre de MQCLOSE | Affiche le nombre d'appels de MQCLOSE. |
| MQINQ et MQSET | Nombre de MQINQ | Affiche le nombre d'appels de MQINQ. |
| | Nombre de MQSET | Affiche le nombre d'appels de MQSET. |
| MQPUT et MQPUT1 | Nombre de MQPUT/MQPUT1 | Affiche le nombre d'appels de MQPUT et MQPUT1. |

Tableau 10. Eléments pour les ressources de statistiques d'utilisation de l'API par file d'attente (suite)


| Tapez | Élément | Description |
|--------------|--|--|
| | Nombre d'octets MQPUT | Affiche le nombre total d'octets de données insérés par des appels de MQPUT et MQPUT1. |
| | Nombre de messages non persistants MQPUT | Affiche le nombre de messages non persistants insérés par MQPUT. |
| | Nombre de messages persistants MQPUT | Affiche le nombre de messages persistants insérés par MQPUT. |
| | Nombre de messages non persistants MQPUT1 | Affiche le nombre de messages non persistants insérés par MQPUT1. |
| | Nombre de messages persistants MQPUT1 | Affiche le nombre de messages persistants insérés par MQPUT1. |
| | Nombre d'octets non persistants | Affiche le nombre d'octets insérés dans des messages non persistants. |
| | Nombre d'octets persistants | Affiche le nombre d'octets insérés dans des messages persistants. |
| | Nombre de commandes put en file d'attente évitées | |
| | Nombre d'octets en file d'attente évités | |
| | contention de verrouillage | |
| MQGET | Nombre de MQGET | |
| | Nombre d'octets MQGET | |
| | Nombre de messages non persistants MQGET destructifs | Nombre de messages non persistants supprimés de la file d'attente par MQGET. |
| | Nombre de messages persistants MQGET destructifs | Nombre de messages persistants supprimés de la file d'attente par MQGET. |
| | Nombre d'octets non persistants MQGET destructifs | Affiche le nombre d'octets de messages non persistants renvoyés à MQGET. |
| | Nombre d'octets persistants MQGET destructifs | Affiche le nombre d'octets de messages persistants renvoyés à MQGET. |

Tableau 10. Eléments pour les ressources de statistiques d'utilisation de l'API par file d'attente (suite)

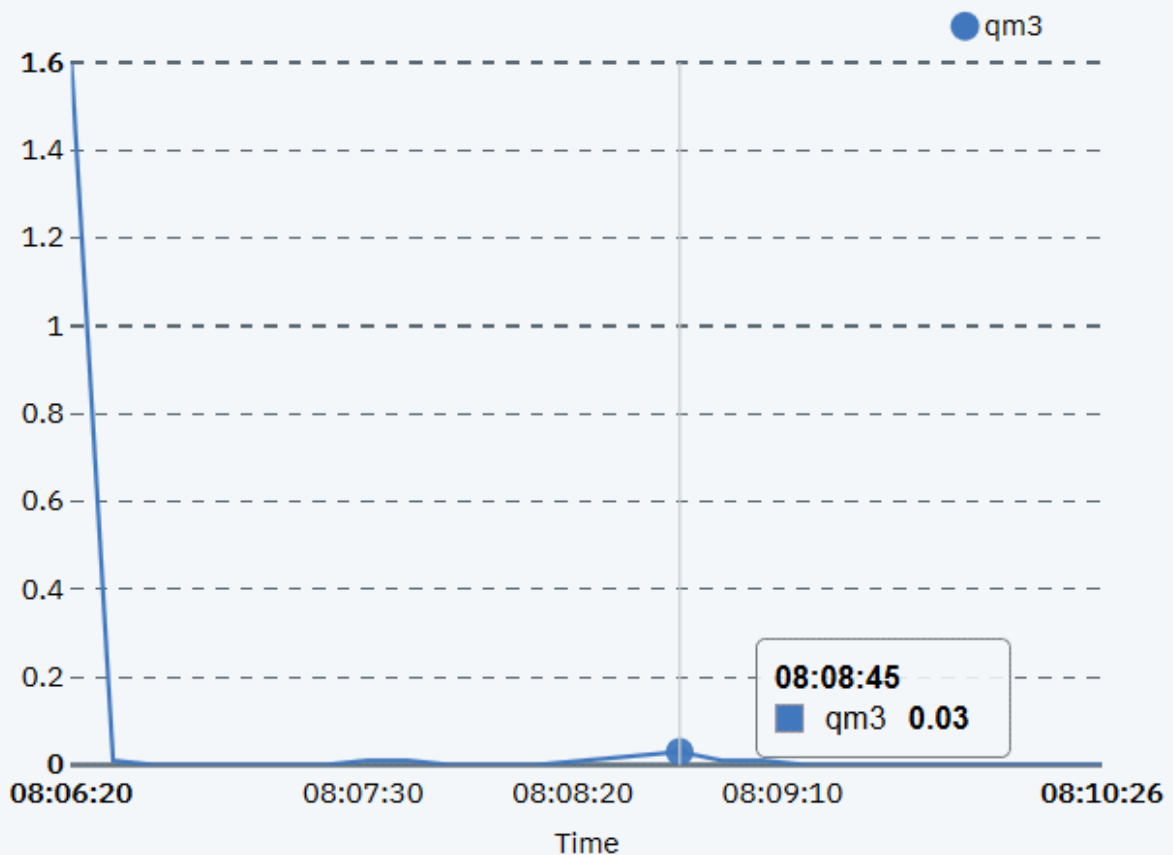
| Tapez | Elément | Description |
|-------|---|--|
| | Nombre de messages non persistants de visualisation MQGET | Affiche le nombre de messages non persistants visualisés. |
| | Nombre de messages persistants de visualisation MQGET | Affiche le nombre de messages persistants visualisés. |
| | Nombre d'octets non persistants de visualisation MQGET | Affiche le nombre d'octets de messages non persistants visualisés. |
| | Nombre d'octets persistants de visualisation MQGET | Affiche le nombre d'octets de messages persistants visualisés. |
| | Messages arrivés à expiration | Affiche le nombre de messages arrivés à expiration. |
| | Nombre de files d'attente purgées | Affiche le nombre de files d'attente qui ont été purgées. |
| | Durée moyenne en file d'attente | |
| | Temps passé en file d'attente | |

- f) Sélectionnez un gestionnaire de files d'attente à surveiller et spécifiez la couleur d'affichage des informations pour ce gestionnaire de files d'attente. Cliquez sur **Ajouter** pour ajouter des gestionnaires de files d'attente supplémentaires. Vous pouvez spécifier jusqu'à cinq gestionnaires de files d'attente.
- g) Cliquer sur **Sauvegarder**.

Résultats

Une fois que vous avez configuré le graphique, vous devez attendre un court moment avant que les données ne s'affichent dans le graphique. Les données sont affichées le long d'un axe temporel. Chaque point de données représente la fin de la période de 10 secondes au cours de laquelle les données sont collectées. Vous pouvez survoler des points de données dans le graphique pour afficher des informations détaillées, comme illustré dans l'exemple suivant: 

User CPU time - percentage estimate for que...




V 9.0.1

Configuration des présentations de tableau de bord

Un tableau de bord est un conteneur dans le IBM MQ Console dans lequel les widgets sont affichés. Vous pouvez créer plusieurs onglets de tableau de bord pour afficher différentes sélections d'informations.

Pourquoi et quand exécuter cette tâche

Vous pouvez configurer chaque onglet de tableau de bord en cliquant sur la flèche en regard du nom

de l'onglet . Vous pouvez modifier le nom de l'onglet et en ajouter une description. Vous pouvez également configurer le nombre de colonnes de l'onglet.

Vous pouvez configurer la présentation des widgets dans un onglet de tableau de bord en les faisant glisser et en les déposant.

Procédure

- «Création et suppression d'onglets de tableau de bord», à la page 133
- «Importation et exportation des agencements de tableau de bord», à la page 133

Création et suppression d'onglets de tableau de bord

Vous pouvez créer automatiquement un onglet de tableau de bord qui affiche des informations sur un gestionnaire de files d'attente local spécifique. Vous pouvez créer et supprimer manuellement des onglets de tableau de bord.

Pourquoi et quand exécuter cette tâche

Lorsque vous créez automatiquement un onglet de tableau de bord pour afficher des informations sur un gestionnaire de files d'attente local spécifique, les widgets suivants sont automatiquement ajoutés:


- Widget Files d'attente
- Widget Canaux de connexion client
- Widget Canaux
- Widget Programmes d'écoute
- Widget Abonnements
- Widget Rubriques
- Widget Informations d'authentification

Procédure

- Pour créer un onglet de tableau de bord:
 - a) Cliquez sur l'icône plus en regard de vos onglets de tableau de bord existants



- b) Entrez un nom pour le nouvel onglet.
 - c) Facultatif : Entrez une description pour le nouvel onglet.
 - d) Cliquez sur **Ajouter**.
- Pour créer automatiquement un onglet de tableau de bord pour un gestionnaire de files d'attente spécifique:
 - a) Sélectionnez le gestionnaire de files d'attente dans le widget de gestionnaire de files d'attente local.
 - b) Sélectionnez ... > **Ajouter un nouvel onglet de tableau de bord**
Un nouvel onglet de tableau de bord est créé. L'onglet contient le nom du gestionnaire de files d'attente.
 - Pour supprimer un onglet de tableau de bord:

- a) Cliquez sur la flèche en regard du nom de l'onglet de tableau de bord  .
- b) Sélectionnez **Supprimer l'onglet**.
- c) Confirmez que vous souhaitez supprimer l'onglet de tableau de bord en cliquant sur **Supprimer**. L'onglet est supprimé.

Importation et exportation des agencements de tableau de bord



Vous pouvez sauvegarder une présentation de tableau de bord en l'exportant à partir du IBM MQ Console. Vous pouvez importer une présentation de tableau de bord sauvegardée dans le IBM MQ Console.

Pourquoi et quand exécuter cette tâche

Lorsque vous exportez un tableau de bord, vous créez un fichier .json sur votre disque local. Par la suite, vous pouvez importer le fichier .json dans un tableau de bord pour recréer la présentation. Lorsque vous importez une présentation de tableau de bord, vous pouvez choisir d'ajouter les onglets importés à une présentation de tableau de bord existante. Vous pouvez également remplacer la présentation de tableau de bord existante par la présentation importée.



Procédure

- Pour exporter un agencement de tableau de bord :

- a) Cliquez sur l'icône de menu du tableau de bord  .
- b) Sélectionnez **Exporter le tableau de bord**.

Le fichier est sauvegardé dans le dossier de téléchargement de votre navigateur.

- Pour importer un agencement de tableau de bord :

- a) Cliquez sur l'icône de menu du tableau de bord  .
- b) Sélectionnez **Importer un tableau de bord**.

La fenêtre Importation d'une configuration de tableau de bord s'ouvre.

- c) Cliquez sur **Parcourir** et recherchez l'emplacement du fichier contenant votre configuration.
- d) Choisissez le mode d'importation des onglets de tableau de bord:
Vous avez le choix entre les différentes options suivantes :

- **Ajouter des onglets de tableau de bord importés à un tableau de bord existant**
- **Remplacer le tableau de bord existant par des onglets de tableau de bord importés**

- e) Cliquez sur **Importer**.

Les onglets du tableau de bord sont importés.







Commandes du tableau de bord

Utilisez les contrôles situés dans la partie supérieure du tableau de bord pour activer le traçage de IBM MQ Console , accéder à l'aide en ligne, afficher des informations sur le IBM MQ Console et se déconnecter du IBM MQ Console.

Pourquoi et quand exécuter cette tâche

Les contrôles de tableau de bord s'appliquent aux fonctions générales de IBM MQ Console.

Procédure

- Accédez à IBM Documentation for IBM MQ en cliquant sur l'icône d'aide .
- Cliquez sur l'icône de menu  pour importer, exporter ou réinitialiser le tableau de bord. Pour plus d'informations, voir [«Configuration des présentations de tableau de bord»](#), à la page 132.
- Cliquez sur l'icône des paramètres  pour activer ou désactiver la trace de diagnostic pour le IBM MQ Console.
- Affichez des informations sur le IBM MQ Console, y compris des informations sur l'utilisateur connecté, en cliquant sur l'icône de personne  et en sélectionnant **A propos de**.

- Déconnectez-vous de IBM MQ Console en cliquant sur **Déconnexion**.
Cette option n'est pas disponible si vous êtes connecté à IBM MQ Console par un certificat client ou si l'exemple de configuration de la sécurité `no_security.xml` est utilisé. Pour plus d'informations sur les configurations de sécurité, voir [Configuration de la sécurité IBM MQ Console](#).

Raccourcis-clavier

Vous pouvez utiliser les raccourcis clavier pour utiliser le IBM MQ Console.

Le tableau suivant répertorie les raccourcis disponibles. Tous les raccourcis sont utilisés à partir d'un widget.

| Tableau 11. Raccourcis clavier de la console | |
|--|--|
| Clé | Action |
| p | Afficher les propriétés de l'objet sélectionné |
| c | Créer un nouvel objet |
| Maj-d | Supprimer l'objet sélectionné |
| a | Sélectionner tous les objets d'un widget |
| Maj-a | Désélectionner les objets |

Windows

Linux

Administration à l'aide de IBM MQ Explorer

IBM MQ Explorer vous permet d'effectuer l'administration locale ou à distance de votre réseau à partir d'un ordinateur exécutant Windows, ou Linux x86-64 uniquement.

IBM MQ for Windows et IBM MQ for Linux x86-64 fournissent une interface d'administration appelée IBM MQ Explorer pour effectuer des tâches d'administration au lieu d'utiliser des commandes de contrôle ou MQSC. [La comparaison des ensembles de commandes](#) vous montre ce que vous pouvez faire à l'aide de l'IBM MQ Explorer.

Le IBM MQ Explorer vous permet d'effectuer une administration locale ou à distance de votre réseau à partir d'un ordinateur exécutant Windows, ou Linux x86-64, en pointant le IBM MQ Explorer vers les gestionnaires de files d'attente et les clusters qui vous intéressent. Il peut se connecter à distance aux gestionnaires de files d'attente s'exécutant sur n'importe quelle plateforme prise en charge, notamment z/OS, ce qui vous permet de visualiser, d'explorer et de modifier l'intégralité de votre réseau principal de messagerie depuis la console.

Pour configurer les gestionnaires de files d'attente IBM MQ distantes afin que IBM MQ Explorer puisse les administrer, voir «Logiciels prérequis et définitions pour IBM MQ Explorer», à la page 137.

Il vous permet d'effectuer des tâches, généralement associées à la configuration et à l'optimisation de l'environnement de travail pour IBM MQ, localement ou à distance dans un domaine système Windows ou Linux x86-64 .

Sous Linux, le démarrage de IBM MQ Explorer peut échouer si vous disposez de plusieurs installations Eclipse . Si cela se produit, démarrez IBM MQ Explorer à l'aide d'un ID utilisateur différent de celui que vous utilisez pour l'autre installation Eclipse .

Sous Linux, pour démarrer correctement IBM MQ Explorer , vous devez pouvoir écrire un fichier dans votre répertoire de base et ce dernier doit exister.

Windows

Linux

Ce que vous pouvez faire avec le IBM MQ Explorer

Vous pouvez utiliser IBM MQ Explorer pour effectuer des tâches d'administration à l'aide d'une série de vues de contenu et de boîtes de dialogue de propriétés. Vous pouvez également étendre le IBM MQ Explorer en écrivant un ou plusieurs plug-in Eclipse .

Tâches IBM MQ Explorer

Avec IBM MQ Explorer, vous pouvez effectuer les tâches suivantes:

- Créez et supprimez un gestionnaire de files d'attente (sur votre machine locale uniquement).
- Démarrez et arrêtez un gestionnaire de files d'attente (sur votre machine locale uniquement).
- Définissez, affichez et modifiez les définitions des objets IBM MQ tels que les files d'attente et les canaux.
- Parcourez les messages d'une file d'attente.
- Démarrer et arrêter un canal.
- Afficher des informations de statut sur un canal, un programme d'écoute, une file d'attente ou des objets de service.
- Afficher les gestionnaires de files d'attente dans un cluster.
- Vérifiez les applications, les utilisateurs ou les canaux pour lesquels une file d'attente particulière est ouverte.
- Créez un cluster de gestionnaires de files d'attente à l'aide de l'assistant de création de cluster.
- Ajoutez un gestionnaire de files d'attente à un cluster à l'aide de l'assistant Ajout d'un gestionnaire de files d'attente à un cluster.
- Gérez l'objet d'informations d'authentification, utilisé avec la sécurité du canal TLS (Transport Layer Security).
- Créer et supprimer des initiateurs de canal, des moniteurs de déclenchement et des programmes d'écoute.
- Démarrez ou arrêtez les serveurs de commandes, les initiateurs de canal, les moniteurs de déclenchement et les programmes d'écoute.
- Définissez des services spécifiques pour qu'ils démarrent automatiquement lorsqu'un gestionnaire de files d'attente est démarré.
- Modifiez les propriétés des gestionnaires de files d'attente.
- Modifiez le gestionnaire de files d'attente par défaut local.
- Appelez l'interface graphique **strmqikm** (ikeyMan) pour gérer les certificats TLS, associer des certificats à des gestionnaires de files d'attente et configurer et configurer des magasins de certificats (sur votre machine locale uniquement).
- Créez des objets JMS à partir d'objets IBM MQ et des objets IBM MQ à partir d'objets JMS .
- Créez une fabrique de connexions JMS pour l'un des types actuellement pris en charge.
- Modifiez les paramètres d'un service, comme le numéro de port TCP d'un programme d'écoute ou le nom d'une file d'attente d'initiateur de canal.
- Démarrez ou arrêtez la trace de service.

Vues de contenu et boîtes de dialogue de propriétés

Vous effectuez des tâches d'administration à l'aide d'une série de vues de contenu et de boîtes de dialogue de propriété.

Vue Contenu

Une vue de contenu est un panneau qui peut afficher les éléments suivants:

- Attributs et options d'administration relatifs à IBM MQ .
- Attributs et options d'administration relatifs à un ou plusieurs objets associés.
- Attributs et options d'administration d'un cluster.

Boîtes de dialogue de propriétés

Une boîte de dialogue de propriété est un panneau qui affiche les attributs relatifs à un objet dans une série de zones, dont certaines peuvent être éditées.

Vous naviguez dans IBM MQ Explorer à l'aide de la vue Navigator . Le Navigator vous permet de sélectionner la vue de contenu dont vous avez besoin.

Extension du IBM MQ Explorer

Le IBM MQ Explorer présente les informations dans un style cohérent avec celui de l'infrastructure Eclipse et les autres applications de plug-in prises en charge par Eclipse .

Grâce à l'extension du IBM MQ Explorer, les administrateurs système ont la possibilité de personnaliser le IBM MQ Explorer afin d'améliorer la manière dont ils administrent IBM MQ.

Pour plus d'informations, voir [Extension de MQ Explorer](#).

Choix de l'utilisation du IBM MQ Explorer

Lorsque vous décidez d'utiliser le IBM MQ Explorer lors de votre installation, prenez en compte les informations répertoriées dans cette rubrique.

Vous devez prendre en compte les points suivants:

Noms d'objet

Si vous utilisez des noms en minuscules pour les gestionnaires de files d'attente et les autres objets avec IBM MQ Explorer, lorsque vous utilisez les objets à l'aide des commandes MQSC, vous devez les placer entre apostrophes ou IBM MQ ne les reconnaît pas.

Grands gestionnaires de files d'attente

IBM MQ Explorer fonctionne mieux avec les petits gestionnaires de files d'attente. Si vous disposez d'un grand nombre d'objets sur un seul gestionnaire de files d'attente, vous risquez de rencontrer des retards lors de l'extraction par IBM MQ Explorer des informations requises à présenter dans une vue.

Groupes

Les clusters IBM MQ peuvent potentiellement contenir des centaines ou des milliers de gestionnaires de files d'attente. La IBM MQ Explorer présente les gestionnaires de files d'attente dans un cluster à l'aide d'une structure arborescente. La taille physique d'un cluster n'affecte pas de manière significative la vitesse du IBM MQ Explorer car IBM MQ Explorer ne se connecte pas aux gestionnaires de files d'attente du cluster tant que vous ne les avez pas sélectionnés.

Configuration de IBM MQ Explorer

Cette section décrit les étapes à suivre pour configurer IBM MQ Explorer.

- [«Logiciels prérequis et définitions pour IBM MQ Explorer»](#), à la page 137
- [«Sécurité de IBM MQ Explorer»](#), à la page 138
- [«Affichage et masquage des gestionnaires de files d'attente et des clusters dans IBM MQ Explorer»](#), à la page 142
- [«Appartenance à un cluster et IBM MQ Explorer»](#), à la page 143
- [«Conversion de données pour IBM MQ Explorer»](#), à la page 143

Logiciels prérequis et définitions pour IBM MQ Explorer

Vérifiez que vous respectez les exigences suivantes avant de tenter d'utiliser le IBM MQ Explorer.

IBM MQ Explorer peut se connecter à des gestionnaires de files d'attente éloignées à l'aide du protocole de communication TCP/IP uniquement.

Vérifiez que:

1. Un serveur de commandes est en cours d'exécution sur chaque gestionnaire de files d'attente administré à distance.
2. Un objet programme d'écoute TCP/IP approprié doit être en cours d'exécution sur chaque gestionnaire de files d'attente éloignées. Cet objet peut être le programme d'écoute IBM MQ ou, sur les systèmes UNIX and Linux , le démon inetd.

3. Un canal de connexion serveur, par défaut nommé SYSTEM.ADMIN.SVRCONN, existe sur tous les gestionnaires de files d'attente éloignées.

Vous pouvez créer le canal à l'aide de la commande MQSC suivante:

```
DEFINE CHANNEL(SYSTEM.ADMIN.SVRCONN) CHLTYPE(SVRCONN)
```

Cette commande crée une définition de canal de base. Si vous souhaitez une définition plus sophistiquée (pour configurer la sécurité, par exemple), vous avez besoin de paramètres supplémentaires. Pour plus d'informations, voir [DEFINE CHANNEL](#).

4. La file d'attente système, SYSTEM.MQEXPLORER.REPLY.MODEL, doit exister.

Sécurité de IBM MQ Explorer

Si vous utilisez IBM MQ dans un environnement où il est important pour vous de contrôler l'accès des utilisateurs à des objets particuliers, vous devrez peut-être prendre en compte les aspects de sécurité de l'utilisation de IBM MQ Explorer.

Autorisation d'utiliser le IBM MQ Explorer

Tout utilisateur peut utiliser le IBM MQ Explorer, mais certains droits sont requis pour la connexion, l'accès et la gestion des gestionnaires de files d'attente.

Pour effectuer des tâches d'administration locales à l'aide de IBM MQ Explorer, un utilisateur doit disposer des droits nécessaires pour effectuer les tâches d'administration. Si l'utilisateur est membre du groupe mqm, il est autorisé à effectuer toutes les tâches d'administration locales.

Pour se connecter à un gestionnaire de files d'attente éloignées et effectuer des tâches d'administration à distance à l'aide de IBM MQ Explorer, l'utilisateur exécutant IBM MQ Explorer doit disposer des droits suivants:

- Droits CONNECT sur l'objet gestionnaire de files d'attente cible
- Droits INQUIRE sur l'objet de gestionnaire de files d'attente cible
- Droits DISPLAY sur l'objet gestionnaire de files d'attente cible
- Droits INQUIRE sur la file d'attente, SYSTEM.MQEXPLORER.REPLY.MODEL
- Droits DISPLAY sur la file d'attente, SYSTEM.MQEXPLORER.REPLY.MODEL
- Droit d'entrée (get) sur la file d'attente, SYSTEM.MQEXPLORER.REPLY.MODEL
- Droit de sortie (put) sur la file d'attente, SYSTEM.MQEXPLORER.REPLY.MODEL
- Droit de sortie (put) sur la file d'attente, SYSTEM.ADMIN.COMMAND.QUEUE
- Droits INQUIRE sur la file d'attente, SYSTEM.ADMIN.COMMAND.QUEUE
- Droit d'exécution de l'action sélectionnée

Remarque : Les droits d'entrée sont liés à l'entrée à l'utilisateur à partir d'une file d'attente (opération d'extraction). Le droit OUTPUT est lié à la sortie de l'utilisateur vers une file d'attente (opération d'insertion).

Pour vous connecter à un gestionnaire de files d'attente éloignées sous IBM MQ for z/OS et effectuer des tâches d'administration à distance à l'aide de IBM MQ Explorer, les informations suivantes doivent être fournies:

- Un profil RACF pour la file d'attente système, SYSTEM.MQEXPLORER.REPLY.MODEL
- Un profil RACF pour les files d'attente, AMQ.MQEXPLORER. *

En outre, l'utilisateur exécutant le IBM MQ Explorer doit disposer des droits suivants:

- RACF droits UPDATE sur la file d'attente système, SYSTEM.MQEXPLORER.REPLY.MODEL
- RACF droits UPDATE sur les files d'attente, AMQ.MQEXPLORER. *
- Droits CONNECT sur l'objet gestionnaire de files d'attente cible

- Droit d'exécution de l'action sélectionnée
- Droits d'accès en lecture (READ) à tous les profils hlq.DISPLAY.object dans la classe MQCMD5

Pour plus d'informations sur l'octroi de droits sur des objets IBM MQ , voir [Attribution de droits d'accès à un objet IBM MQ sur des systèmes UNIX ou Linux et Windows](#) .

Si un utilisateur tente d'effectuer une opération qu'il n'est pas autorisé à effectuer, le gestionnaire de files d'attente cible appelle des procédures d'échec d'autorisation et l'opération échoue.

Le filtre par défaut dans IBM MQ Explorer consiste à afficher tous les objets IBM MQ . S'il existe des objets IBM MQ sur lesquels un utilisateur ne dispose pas des droits DISPLAY, des échecs d'autorisation sont générés. Si des événements de droits sont enregistrés, limitez la plage d'objets affichés aux objets sur lesquels l'utilisateur dispose des droits DISPLAY.

Sécurité pour la connexion à des gestionnaires de files d'attente éloignées à partir de IBM MQ Explorer

Vous devez sécuriser le canal entre IBM MQ Explorer et chaque gestionnaire de files d'attente éloignées.

IBM MQ Explorer se connecte aux gestionnaires de files d'attente éloignées en tant qu'application client MQI. Cela signifie que chaque gestionnaire de files d'attente éloignées doit disposer d'une définition de canal de connexion serveur et d'un programme d'écoute TCP/IP approprié. Si vous ne sécurisez pas votre canal de connexion serveur, une application malveillante peut se connecter au même canal de connexion serveur et accéder aux objets du gestionnaire de files d'attente avec des droits illimités. Pour sécuriser votre canal de connexion serveur, indiquez une valeur non vide pour l'attribut MCAUSER du canal, utilisez des enregistrements d'authentification de canal ou utilisez un exit de sécurité.

La valeur par défaut de l'attribut MCAUSER est l'ID utilisateur local. Si vous indiquez un nom d'utilisateur non vide comme attribut MCAUSER du canal de connexion serveur, tous les programmes qui se connectent au gestionnaire de files d'attente à l'aide de ce canal s'exécutent avec l'identité de l'utilisateur nommé et disposent du même niveau de droits d'accès. Cela ne se produit pas si vous utilisez des enregistrements d'authentification de canal.

Utilisation d'un exit de sécurité avec IBM MQ Explorer

Vous pouvez spécifier un exit de sécurité par défaut et des exits de sécurité spécifiques au gestionnaire de files d'attente à l'aide de IBM MQ Explorer.

Vous pouvez définir un exit de sécurité par défaut, qui peut être utilisé pour toutes les nouvelles connexions client à partir de IBM MQ Explorer. Cet exit par défaut peut être remplacé lors de l'établissement d'une connexion. Vous pouvez également définir un exit de sécurité pour un seul gestionnaire de files d'attente ou un ensemble de gestionnaires de files d'attente, qui prend effet lorsqu'une connexion est établie. Vous spécifiez des exits à l'aide de IBM MQ Explorer. Pour plus d'informations, reportez-vous à l'aide d'IBM MQ Explorer.

Utilisation de IBM MQ Explorer pour la connexion à un gestionnaire de files d'attente éloignées à l'aide de canaux MQI TLS

IBM MQ Explorer se connecte aux gestionnaires de files d'attente éloignées à l'aide d'un canal MQI. Si vous souhaitez sécuriser le canal MQI à l'aide de la sécurité TLS, vous devez établir le canal à l'aide d'une table de définition de canal du client.

Pour plus d'informations sur l'établissement d'un canal MQI à l'aide d'une table de définition de canal du client, voir [Présentation d' IBM MQ MQI clients](#).

Une fois que vous avez établi le canal à l'aide d'une table de définition de canal du client, vous pouvez utiliser IBM MQ Explorer pour vous connecter à un gestionnaire de files d'attente éloignées à l'aide d'un canal MQI activé par TLS, comme décrit dans [«Tâches sur le système qui héberge le gestionnaire de files d'attente éloignées»](#), à la page 139 et [«Tâches sur le système qui héberge le IBM MQ Explorer»](#), à la page 140.

Tâches sur le système qui héberge le gestionnaire de files d'attente éloignées

Sur le système hébergeant le gestionnaire de files d'attente éloignées, effectuez les tâches suivantes:

1. Définissez une paire de canaux de connexion serveur et de connexion client et spécifiez la valeur appropriée pour l'attribut *SSLCIPH* sur la connexion serveur sur les deux canaux. Pour plus d'informations sur l'attribut *SSLCIPH*, voir [Protection des canaux avec TLS](#).
2. Envoyez la table de définition de canal AMQCLCHL .TAB, qui se trouve dans le répertoire @ipcc du gestionnaire de files d'attente, au système hébergeant IBM MQ Explorer.
3. Démarrez un programme d'écoute TCP/IP sur un port désigné.
4. Placez les certificats de l'autorité de certification et les certificats TLS personnels dans le répertoire SSL du gestionnaire de files d'attente:
 - /var/mqm/qmgrs/+QMNAME+/SSL pour les systèmes UNIX and Linux
 - C:\Program Files\IBM\MQ\qmgrs\+QMNAME+\SSL pour les systèmes Windows
 Où +QMNAME+ est un jeton représentant le nom du gestionnaire de files d'attente.
5. Créez un fichier de base de données de clés de type CMS nommé key .kdb. Stockez le mot de passe dans un fichier en cochant l'option dans le fichier **strmqikm** (iKeyman) Interface graphique, ou en utilisant l'option -stash avec les commandes **runmqckm**.
6. Ajoutez les certificats de l'autorité de certification à la base de données de clés créée à l'étape précédente.
7. Importez le certificat personnel du gestionnaire de files d'attente dans la base de données de clés.

Pour plus d'informations sur l'utilisation de TLS sur les systèmes Windows, voir [Utilisation de TLS sur UNIX, Linux, and Windows](#).

Tâches sur le système qui héberge le IBM MQ Explorer

Sur le système hébergeant le IBM MQ Explorer, effectuez les tâches suivantes:

1. Créez un fichier de base de données de clés de type JKS nommé key .jks. Définissez un mot de passe pour ce fichier de base de données de clés.

Le IBM MQ Explorer utilise les fichiers de clés Java (JKS) pour la sécurité TLS. Le fichier de clés créé pour la configuration de TLS pour IBM MQ Explorer doit donc correspondre à celui-ci.
2. Ajoutez les certificats de l'autorité de certification à la base de données de clés créée à l'étape précédente.
3. Importez le certificat personnel du gestionnaire de files d'attente dans la base de données de clés.
4. Sur les systèmes Windows et Linux, démarrez IBM MQ Explorer à l'aide du menu système, du fichier exécutable MQExplorer ou de la commande **strmqcfcg**.
5. Dans la barre d'outils IBM MQ Explorer, cliquez sur **Fenêtre-> Préférences**, puis développez **IBM MQ Explorer** et cliquez sur **Magasins de certificats client SSL**. Entrez le nom et le mot de passe du fichier JKS créé à l'étape 1 de «Tâches sur le système qui héberge le IBM MQ Explorer», à la page 140, dans le magasin de certificats de confiance et dans le magasin de certificats personnels, puis cliquez sur **OK**.
6. Fermez la fenêtre **Préférences** et cliquez avec le bouton droit de la souris sur **Gestionnaires de files d'attente**. Cliquez sur **Afficher / Masquer les gestionnaires de files d'attente**, puis sur **Ajouter** dans l'écran **Afficher / Masquer les gestionnaires de files d'attente**.
7. Entrez le nom du gestionnaire de files d'attente et sélectionnez l'option **Se connecter directement**. Cliquez sur **Suivant**.
8. Sélectionnez **Utiliser la table de définition de canal du client (CCDT)** et indiquez l'emplacement du fichier de table de canaux que vous avez transféré du gestionnaire de files d'attente éloignées à l'étape 2 dans «Tâches sur le système qui héberge le gestionnaire de files d'attente éloignées», à la page 139 sur le système hébergeant le gestionnaire de files d'attente éloignées.
9. Cliquez sur **Finir**. Vous pouvez maintenant accéder au gestionnaire de files d'attente éloignées à partir de la IBM MQ Explorer.

Connexion via un autre gestionnaire de files d'attente avec IBM MQ Explorer

IBM MQ Explorer vous permet de vous connecter à un gestionnaire de files d'attente via un gestionnaire de files d'attente intermédiaire, auquel IBM MQ Explorer est déjà connecté.

Dans ce cas, IBM MQ Explorer insère des messages de commande PCF dans le gestionnaire de files d'attente intermédiaire, en spécifiant ce qui suit:

- Le paramètre *ObjectQMGrName* dans le descripteur d'objet (MQOD) comme nom du gestionnaire de files d'attente cible. Pour plus d'informations sur la résolution de nom de file d'attente, voir [Résolution de nom](#).
- Le paramètre *UserIdentifier* dans le descripteur de message (MQMD) en tant qu' *userIdlocal*.

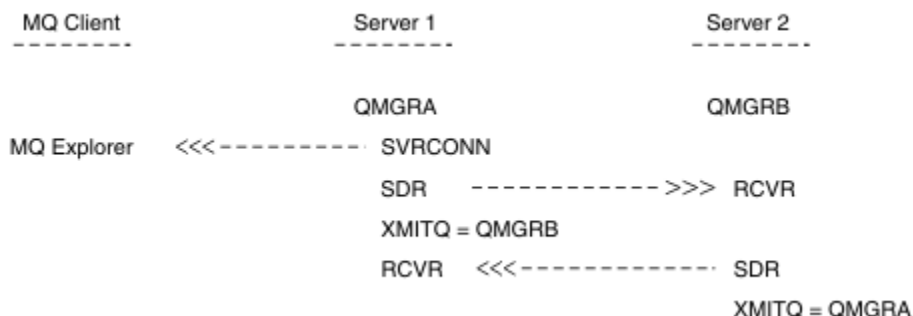
Si la connexion est ensuite utilisée pour se connecter au gestionnaire de files d'attente cible via un gestionnaire de files d'attente intermédiaire, *userId* est de nouveau transmis dans le paramètre *UserIdentifier* du descripteur de message (MQMD). Pour que le programme d'écoute MCA sur le gestionnaire de files d'attente cible accepte ce message, l'attribut MCAUSER doit être défini ou l' *userId* doit déjà exister avec le droit d'insertion.

Le serveur de commandes du gestionnaire de files d'attente cible insère des messages dans la file d'attente de transmission en spécifiant *userId* dans le paramètre *UserIdentifier* du descripteur de message (MQMD). Pour que cette insertion aboutisse, l' *userId* doit déjà exister sur le gestionnaire de files d'attente cible avec les droits d'insertion.

L'exemple suivant montre comment connecter un gestionnaire de files d'attente, via un gestionnaire de files d'attente intermédiaire, à IBM MQ Explorer.

Etablissez une connexion d'administration à distance à un gestionnaire de files d'attente. Vérifiez que:

- Le gestionnaire de files d'attente sur le serveur est actif et un canal de connexion serveur (SVRCONN) est défini.
- Le programme d'écoute est actif.
- Le serveur de commandes est actif.
- SYSTEME.SYSTEM.MQ.EXPLORER.REPLY.MODEL a été créée et vous disposez des droits suffisants.
- Les programmes d'écoute du gestionnaire de files d'attente, les serveurs de commandes et les canaux émetteurs sont démarrés.



Dans cet exemple :

- IBM MQ Explorer est connecté au gestionnaire de files d'attente QMGRA (exécuté sur Server1) à l'aide d'une connexion client.
- Le gestionnaire de files d'attente QMGRB sur Server2 peut désormais être connecté à IBM MQ Explorer via un gestionnaire de files d'attente intermédiaire (QMGRA)
- Lors de la connexion à QMGRB avec IBM MQ Explorer, sélectionnez QMGRA comme gestionnaire de files d'attente intermédiaire

Dans cette situation, il n'y a pas de connexion directe à QMGRB à partir de IBM MQ Explorer; la connexion à QMGRB s'effectue via QMGRA.

Le gestionnaire de files d'attente QMGRB sur Server2 est connecté à QMGRA sur Server1 à l'aide de canaux émetteurs-récepteurs. Le canal entre QMGRA et QMGRB doit être configuré de sorte que l'administration à distance soit possible ; voir «Préparation des canaux et des files d'attente de transmission pour l'administration à distance», à la page 202.

Affichage et masquage des gestionnaires de files d'attente et des clusters dans IBM MQ Explorer

IBM MQ Explorer peut afficher plusieurs gestionnaires de files d'attente à la fois. Dans le panneau Afficher / Masquer le gestionnaire de files d'attente (sélectionnable dans le menu du noeud d'arborescence des gestionnaires de files d'attente), vous pouvez choisir d'afficher ou non des informations sur une autre machine (distante). Les gestionnaires de files d'attente locaux sont détectés automatiquement.

Pour afficher un gestionnaire de files d'attente éloignées:

1. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence **Gestionnaires de files d'attente** , puis sélectionnez **Afficher / Masquer les gestionnaires de files d'attente**.
2. Cliquez sur **Ajouter**. Le panneau Afficher / Masquer les gestionnaires de files d'attente s'affiche.
3. Entrez le nom du gestionnaire de files d'attente éloignées et le nom d'hôte ou l'adresse IP dans les zones fournies.

Le nom d'hôte ou l'adresse IP est utilisé pour établir une connexion client au gestionnaire de files d'attente éloignées à l'aide de son canal de connexion serveur par défaut, SYSTEM.ADMIN.SVRCONN, ou un canal de connexion serveur défini par l'utilisateur.

4. Cliquez sur **Terminer**.

Le panneau Afficher / Masquer les gestionnaires de files d'attente affiche également la liste de tous les gestionnaires de files d'attente visibles. Vous pouvez utiliser ce panneau pour masquer les gestionnaires de files d'attente dans la vue de navigation.

Si IBM MQ Explorer affiche un gestionnaire de files d'attente qui est membre d'un cluster, le cluster est détecté et affiché automatiquement.

Pour exporter la liste des gestionnaires de files d'attente éloignées à partir de ce panneau:

1. Fermez le panneau Afficher / Masquer les gestionnaires de files d'attente.
2. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence **IBM MQ** le plus élevé dans la sous-fenêtre de navigation du IBM MQ Explorer, puis sélectionnez **Exporter les paramètres IBM MQ Explorer** .
3. Cliquez sur **IBM MQ Explorer > IBM MQ Explorer Paramètres**
4. Sélectionnez **Informations de connexion > Gestionnaires de files d'attente éloignées**.
5. Sélectionnez un fichier dans lequel stocker les paramètres exportés.
6. Enfin, cliquez sur **Terminer** pour exporter les informations de connexion du gestionnaire de files d'attente éloignées dans le fichier spécifié.

Pour importer une liste de gestionnaires de files d'attente éloignées:

1. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence **IBM MQ** le plus élevé dans la sous-fenêtre de navigation du IBM MQ Explorer, puis sélectionnez **Importer les paramètres IBM MQ Explorer** .
2. Cliquez sur **IBM MQ Explorer > IBM MQ Explorer Paramètres**
3. Cliquez sur **Parcourir** et accédez au chemin du fichier contenant les informations de connexion du gestionnaire de files d'attente éloignées.
4. Cliquez sur **Ouvrir**. Si le fichier contient une liste de gestionnaires de files d'attente éloignées, la case **Informations de connexion > Gestionnaires de files d'attente éloignées** est cochée.
5. Enfin, cliquez sur **Terminer** pour importer les informations de connexion du gestionnaire de files d'attente éloignées dans le IBM MQ Explorer.

Appartenance à un cluster et IBM MQ Explorer

IBM MQ Explorer requiert des informations sur les gestionnaires de files d'attente qui sont membres d'un cluster.

Si un gestionnaire de files d'attente est membre d'un cluster, le noeud de l'arborescence de clusters est renseigné automatiquement.

Si les gestionnaires de files d'attente deviennent membres de clusters alors que le IBM MQ Explorer est en cours d'exécution, vous devez gérer le IBM MQ Explorer avec des données d'administration à jour sur les clusters afin qu'il puisse communiquer efficacement avec eux et afficher des informations de cluster correctes lorsque cela est demandé. Pour ce faire, IBM MQ Explorer a besoin des informations suivantes:

- Nom d'un gestionnaire de files d'attente de référentiel
- Nom de connexion du gestionnaire de files d'attente de référentiel s'il se trouve sur un gestionnaire de files d'attente éloignées

Avec ces informations, IBM MQ Explorer peut:

- Utilisez le gestionnaire de files d'attente de référentiel pour obtenir la liste des gestionnaires de files d'attente du cluster.
- Administrez les gestionnaires de files d'attente qui sont membres du cluster et qui se trouvent sur des plateformes et des niveaux de commande pris en charge.

L'administration n'est pas possible si:

- Le référentiel choisi devient indisponible. IBM MQ Explorer ne bascule pas automatiquement vers un autre référentiel.
- Le référentiel choisi ne peut pas être contacté via TCP/IP.
- Le référentiel choisi s'exécute sur un gestionnaire de files d'attente qui s'exécute sur une plateforme et dont le niveau de commande n'est pas pris en charge par IBM MQ Explorer.

Les membres du cluster qui peuvent être administrés peuvent être locaux ou distants s'ils peuvent être contactés à l'aide de TCP/IP. IBM MQ Explorer se connecte aux gestionnaires de files d'attente locaux qui sont membres d'un cluster directement, sans utiliser de connexion client.

Conversion de données pour IBM MQ Explorer

IBM MQ Explorer fonctionne avec le CCSID 1208 (UTF-8). Cela permet à IBM MQ Explorer d'afficher correctement les données des gestionnaires de files d'attente éloignées. Que vous vous connectiez à un gestionnaire de files d'attente directement ou à l'aide d'un gestionnaire de files d'attente intermédiaire, IBM MQ Explorer requiert la conversion de tous les messages entrants au CCSID 1208 (UTF-8).

Un message d'erreur est émis si vous tentez d'établir une connexion entre IBM MQ Explorer et un gestionnaire de files d'attente dont le CCSID n'est pas reconnu par IBM MQ Explorer .

Les conversions prises en charge sont décrites dans la rubrique [Conversion de page de codes](#).

Windows Utilisation de l'application IBM MQ Taskbar (Windows uniquement)

L'application IBM MQ Taskbar affiche une icône dans la barre des tâches système Windows sur le serveur. L'icône vous fournit le statut en cours de IBM MQ et un menu à partir duquel vous pouvez effectuer des actions simples.

Sous Windows, l'icône IBM MQ se trouve dans la barre des tâches système du serveur et est remplacée par un symbole de statut codé par des couleurs, qui peut avoir l'une des significations suivantes:

Vert

Fonctionnement correct ; aucune alerte à l'heure actuelle

Bleu

Indéterminé ; IBM MQ est en cours de démarrage ou d'arrêt

Jaune

Alerte ; un ou plusieurs services échouent ou ont déjà échoué

Pour afficher le menu, cliquez avec le bouton droit de la souris sur l'icône IBM MQ . Dans le menu, vous pouvez effectuer les actions suivantes:

- Cliquez sur **Ouvrir** pour ouvrir le moniteur d'alertes IBM MQ .
- Cliquez sur **Quitter** pour quitter l'application IBM MQ Taskbar.
- Cliquez sur **IBM MQ Explorer** pour démarrer le IBM MQ Explorer.
- Cliquez sur **Arrêter IBM MQ** pour arrêter IBM MQ.
- Cliquez sur **A propos de IBM MQ** pour afficher des informations sur le moniteur d'alertes IBM MQ .

L'application du moniteur d'alertes IBM MQ (Windows uniquement)

Le moniteur d'alertes IBM MQ est un outil de détection des erreurs qui identifie et enregistre les problèmes liés à IBM MQ sur une machine locale.

Le moniteur d'alertes affiche des informations sur le statut en cours de l'installation locale d'un serveur IBM MQ . Il surveille également l'interface ACPI (Windows Advanced Configuration and Power Interface) et s'assure que les paramètres ACPI sont appliqués.

A partir du moniteur d'alertes IBM MQ , vous pouvez:

- Accès direct à IBM MQ Explorer
- Afficher les informations relatives à toutes les alertes en attente
- Arrêt du service IBM MQ sur la machine locale
- Acheminer les messages d'alerte sur le réseau vers un compte utilisateur configurable ou vers un poste de travail ou un serveur Windows

Administration des objets IBM MQ locaux

Vous pouvez administrer des objets IBM MQ locaux pour prendre en charge les programmes d'application qui utilisent l'interface MQI (Message Queue Interface).

Pourquoi et quand exécuter cette tâche

Dans ce contexte, l'administration locale implique la création, l'affichage, la modification, la copie et la suppression d'objets IBM MQ .

Outre les approches décrites dans cette section, vous pouvez utiliser IBM MQ Explorer pour administrer des objets IBM MQ locaux. Pour plus d'informations, voir [«Administration à l'aide de IBM MQ Explorer»](#), à la page 135.

Procédure

- Utilisez les informations des rubriques suivantes pour vous aider à administrer les objets IBM MQ locaux.
 - [Programmes d'application utilisant l'interface MQI](#)
 - [«Administration de MQ à l'aide des commandes MQSC»](#), à la page 10
 - [«Utilisation des gestionnaires de files d'attente»](#), à la page 149
 - [«Utilisation des files d'attente locales»](#), à la page 150
 - [«Utilisation de files d'attente alias»](#), à la page 156
 - [«Utilisation des files d'attente modèles»](#), à la page 178
 - [«Utilisation des services»](#), à la page 186

Démarrage et arrêt d'un gestionnaire de files d'attente

Introduction à l'arrêt et au démarrage d'un gestionnaire de files d'attente avec des liens vers des informations supplémentaires.

Pourquoi et quand exécuter cette tâche

Vous pouvez démarrer et arrêter un gestionnaire de files d'attente à l'aide des commandes suivantes:

- Pour démarrer un gestionnaire de files d'attente, utilisez la commande **strmqm**.
- Pour arrêter un gestionnaire de files d'attente, utilisez la commande **endmqm**, qui permet d'arrêter un gestionnaire de files d'attente de trois manières: un arrêt contrôlé ou mis au repos, un arrêt immédiat et un arrêt préventif.

Windows **Linux** Vous pouvez également, sous Windows et Linux, utiliser la IBM MQ Explorer pour démarrer et arrêter un gestionnaire de files d'attente.

Windows Sous Windows, vous pouvez configurer un gestionnaire de files d'attente pour qu'il démarre automatiquement lorsque le système démarre à l'aide de la IBM MQ Explorer.

Procédure

1. Pour plus d'informations sur le démarrage d'un gestionnaire de files d'attente, voir [Démarrage d'un gestionnaire de files d'attente](#).
2. Pour plus d'informations sur l'arrêt d'un gestionnaire de files d'attente, voir [Arrêt d'un gestionnaire de files d'attente](#).

ULW Arrêt manuel d'un gestionnaire de files d'attente

Si les méthodes standard d'arrêt et de suppression d'un gestionnaire de files d'attente échouent, vous pouvez essayer d'arrêter le gestionnaire de files d'attente manuellement.

Pourquoi et quand exécuter cette tâche

La méthode standard d'arrêt des gestionnaires de files d'attente consiste à utiliser la commande **endmqm**, comme décrit dans [Arrêt d'un gestionnaire de files d'attente](#). Si vous ne parvenez pas à arrêter un gestionnaire de files d'attente de manière standard, vous pouvez essayer d'arrêter un gestionnaire de files d'attente manuellement. La manière dont vous effectuez cette opération dépend de la plateforme que vous utilisez.

Procédure

- **Windows**
Pour arrêter un gestionnaire de files d'attente sous Windows, voir [«Arrêt manuel d'un gestionnaire de files d'attente sous Windows»](#), à la page 146.
- **Linux** **UNIX**
Pour arrêter un gestionnaire de files d'attente sous UNIX ou Linux, voir [«Arrêt manuel d'un gestionnaire de files d'attente sous UNIX et Linux»](#), à la page 147.

Information associée

[Création et gestion des gestionnaires de files d'attente sur Multiplatforms endmqm](#)

Windows Arrêt manuel d'un gestionnaire de files d'attente sous Windows

Si vous ne parvenez pas à arrêter un gestionnaire de files d'attente sous Windows à l'aide de la commande **endmqm**, vous pouvez essayer d'arrêter le gestionnaire de files d'attente manuellement en arrêtant tous les processus en cours d'exécution et en arrêtant le service IBM MQ.

Pourquoi et quand exécuter cette tâche

Conseil : Le gestionnaire de tâches Windows et la commande **tasklist** fournissent des informations limitées sur les tâches. Pour plus d'informations permettant de déterminer quels processus sont liés à un gestionnaire de files d'attente particulier, utilisez un outil tel que *Process Explorer* (`procepx.exe`), qui peut être téléchargé à partir du site Web Microsoft à l'adresse <https://www.microsoft.com>.

Pour arrêter un gestionnaire de files d'attente sous Windows, procédez comme suit.

Procédure

1. Répertoriez les noms (ID) des processus en cours d'exécution à l'aide du gestionnaire de tâches Windows.
2. Arrêtez les processus à l'aide du gestionnaire de tâches Windows ou de la commande **taskkill**, dans l'ordre suivant (s'ils sont en cours d'exécution):

| Nom du processus | Description |
|------------------|--|
| AMQZMUC0 | Gestionnaire de processus critique |
| AMQZXMA0 | Contrôleur d'exécution |
| AMQZFUMA | Processus OAM |
| AMQZLAA0 | Agents LQM |
| AMQZLSA0 | Agents LQM |
| AMQZMUFO | Gestionnaire d'utilitaire |
| AMQZMGR0 | Ctrlur de processus |
| AMQZMUR0 | Gestionnaire de processus redémarrable |
| AMQFQPUB | Processus de publication / d'abonnement |
| AMQFCXBA | Processus de travail de courtier |
| AMQRMPPA | Processus de regroupement de processus |
| amqcrsta | Processus de travail de répondeur non à unités d'exécution |
| AMQCRS6B | Connexion client et canal récepteur LU62 |
| AMQRRMFA | Processus de référentiel (pour les clusters) |
| AMQPCSEA | Serveur de commandes |
| runmqtrm | Appel d'un moniteur de déclenchement pour un serveur |
| runmqdlq | Appeler le gestionnaire de files d'attente de rebut |
| runmqchi | Processus de l'initiateur de canal |
| runmqlsr | Processus du programme d'écoute de canal |
| AMQXSSVN | Serveurs de mémoire partagée |

3. Arrêtez le service IBM MQ à partir des **outils d'administration** > **Services** du panneau de configuration Windows .
4. Si vous avez essayé toutes les méthodes et que le gestionnaire de files d'attente ne s'est pas arrêté, réamorçez votre système.

Linux > UNIX Arrêt manuel d'un gestionnaire de files d'attente sous UNIX et Linux

Si vous ne parvenez pas à arrêter un gestionnaire de files d'attente sous UNIX et Linux à l'aide de la commande **endmqm** , vous pouvez essayer d'arrêter le gestionnaire de files d'attente manuellement en arrêtant tous les processus en cours d'exécution et en arrêtant le service IBM MQ .

Pourquoi et quand exécuter cette tâche

Pour arrêter un gestionnaire de files d'attente sous UNIX et Linux, procédez comme suit.

Si vous arrêtez le gestionnaire de files d'attente manuellement, FFST peut être utilisé et les fichiers FDC placés dans `/var/mqm/errors`. Il ne doit pas être considéré comme un incident dans le gestionnaire de files d'attente.

Le gestionnaire de files d'attente redémarre normalement, même après que vous l'ayez arrêté à l'aide de cette méthode d'arrêt manuel.

Procédure

1. Recherchez les ID de processus des programmes du gestionnaire de files d'attente qui sont toujours en cours d'exécution à l'aide de la commande **ps** .

Par exemple, si le gestionnaire de files d'attente est appelé QMNAME, utilisez la commande suivante:

```
ps -ef | grep QMNAME
```

2. Arrêtez tous les processus de gestionnaire de files d'attente qui sont toujours en cours d'exécution à l'aide de la commande **kill** , en spécifiant les PID détectés à l'aide de la commande **ps** .

Pour mettre fin à un processus, utilisez **kill -KILL <pid>** ou la commande **kill -9 <pid>** équivalente.

Vous devez utiliser les PID que vous voulez tuer, un par un, en émettant cette commande à chaque fois.

Important : Si vous utilisez un signal autre que **9(SIGKILL)** , le processus ne s'arrêtera probablement pas et vous obtiendrez des résultats imprévisibles.

Arrêtez les processus dans l'ordre suivant:

| Nom du processus | Description |
|------------------|--|
| amqzmuc0 | Gestionnaire de processus critique |
| amqzxma0 | Contrôleur d'exécution |
| amqzfuma | Processus OAM |
| amqzlaa0 | Agents LQM |
| amqzlsa0 | Agents LQM |
| amqzmuf0 | Gestionnaire d'utilitaire |
| amqzmur0 | Gestionnaire de processus redémarrable |
| amqzmgr0 | Ctrlur de processus |

| Tableau 13. Processus UNIX et Linux à arrêter en cas d'exécution (suite) | |
|--|--|
| Nom du processus | Description |
| amqfqpub | Processus de publication / d'abonnement |
| amqfcxba | Processus de travail de courtier |
| amqrmppa | Processus de regroupement de processus |
| amqcrsta | Processus de travail de répondeur non à unités d'exécution |
| amqcrs6b | Connexion client et canal récepteur LU62 |
| amqrrmfa | Processus de référentiel (pour les clusters) |
| amqpcsea | Serveur de commandes |
| runmqtrm | Appel d'un moniteur de déclenchement pour un serveur |
| runmqdlq | Appeler le gestionnaire de files d'attente de rebut |
| runmqchi | Processus de l'initiateur de canal |
| runmqlsr | Processus du programme d'écoute de canal |

Remarque : Vous pouvez utiliser la commande **kill -9** pour arrêter les processus dont l'arrêt a échoué.

Arrêt des canaux MQI

Lorsque vous émettez une commande STOP CHANNEL sur un canal de connexion serveur, vous pouvez choisir la méthode à utiliser pour arrêter le canal de connexion client. Cela signifie qu'un canal client établissant un appel d'attente MQGET peut être contrôlé et que vous pouvez décider comment et quand arrêter le canal.

La commande STOP CHANNEL peut être exécutée selon trois méthodes, en indiquant comment le canal doit être arrêté :

Quiesce (mise au repos)

Arrête le canal une fois que tous les messages en cours ont été traités.

Si le partage de conversations est activé, IBM MQ MQI client prend connaissance de la demande d'arrêt dans les délais ; cette durée dépend de la vitesse du réseau. L'application client prend connaissance de la demande d'arrêt suite à l'émission d'un appel ultérieur à IBM MQ.

Force

Arrête le canal immédiatement.

Terminer

Arrête le canal immédiatement. Si le canal est en cours d'exécution sous la forme d'un processus, il peut mettre fin au processus du canal ou, si le canal est en cours d'exécution sous la forme d'une unité d'exécution, à son unité d'exécution.

Il s'agit d'un processus à plusieurs phases. Si le mode terminate est utilisé, une tentative d'arrêt du canal de connexion serveur est effectuée, d'abord en mode quiesce, puis en mode force et, si nécessaire, en mode terminate. Le client peut recevoir des codes retour différents pendant les différentes phases d'arrêt. Si le processus ou l'unité d'exécution est arrêtée, le client reçoit une erreur de communication.

Les codes retour renvoyés à l'application varient selon l'appel MQI émis et la commande STOP CHANNEL exécutée. Le client reçoit un code retour MQRC_CONNECTION_QUIESCING ou MQRC_CONNECTION_BROKEN. Si un client détecte MQRC_CONNECTION_QUIESCING, il doit tenter de terminer la transaction en cours et de s'arrêter. Cela n'est pas possible avec

MQRC_CONNECTION_BROKEN. Si le client ne termine pas la transaction et ne s'arrête pas suffisamment vite, il reçoit le code retour CONNECTION_BROKEN au bout de quelques secondes. Une commande STOP CHANNEL avec MODE(FORCE) ou MODE(TERMINATE) est plus susceptible de générer un code CONNECTION_BROKEN qu'avec MODE(QUIESCE).

Information associée

[Canaux](#)

Utilisation des gestionnaires de files d'attente

Exemples de commandes MQSC que vous pouvez utiliser pour afficher ou modifier des attributs de gestionnaire de files d'attente.

Affichage des attributs du gestionnaire de files d'attente

Pour afficher les attributs du gestionnaire de files d'attente spécifié dans la commande **runmqsc**, utilisez la commande **DISPLAY QMGR MQSC**:

```
DISPLAY QMGR
```

La sortie standard de cette commande est illustrée dans la Figure 16, à la page 149

```
DISPLAY QMGR
  1 : DISPLAY QMGR
AMQ8408: Display Queue Manager details.
QMNAME(QM1)
ACCTINT(1800)
ACCTQ(OFF)
ACTVCONO(DISABLED)
ALTDATE(2012-05-27)
AUTHOREV(DISABLED)
CHAD(DISABLED)
CHADEXIT( )
CLWLDATA( )
CLWLEN(100)
CLWLUSEQ(LOCAL)
CMDLEVEL(800)
CONFIGEV(DISABLED)
CRTIME(16.14.01)
DEFXMITQ( )
DISTL(YES)
IPADDRV(IPV4)
LOGGEREV(DISABLED)
MAXHANDS(256)
MAXPROPL(NOLIMIT)
MAXUMSGS(10000)
MONCHL(OFF)
PARENT( )
PLATFORM(WINDOWSNT)
PSNPMMSG(DISCARD)
PSSYNCP( IFFPER)
PSMODE(ENABLED)
REPOS( )
ROUTEREC(MSG)
SCMDSERV(QMGR)
SSLCRYP( )
SSLFIPS(NO)
MQ\Data\qmgrs\QM1\ssl\key)
SSLKEYC(0)
STATCHL(OFF)
STATMQI(OFF)
STRSTPEV(ENABLED)
TREELIFE(1800)
ACCTCONO(DISABLED)
ACCTMQI(OFF)
ACTIVREC(MSG)
ACTVTRC(OFF)
ALTTIME(16.14.01)
CCSID(850)
CHADEV(DISABLED)
CHLEV(DISABLED)
CLWLXIT( )
CLWLMRUC(999999999)
CMDEV(DISABLED)
COMMANDQ(SYSTEM.ADMIN.COMMAND.QUEUE)
CRDATE(2011-05-27)
DEADQ( )
DESCR( )
INHIBTEV(DISABLED)
LOCALEV(DISABLED)
MARKINT(5000)
MAXMSGL(4194304)
MAXPRTY(9)
MONACLS(QMGR)
MONQ(OFF)
PERFMEV(DISABLED)
  PSRTYCNT(5)
PSNMPRES(NORMAL)
QMID(QM1_2011-05-27_16.14.01)
REMOTEEV(DISABLED)
REPOSNL( )
SCHINIT(QMGR)
SSLCRLNL( )
SSLEV(DISABLED)
SSLKEYR(C:\Program Files\IBM\WebSphere
STATACLS(QMGR)
STATINT(1800)
STATQ(OFF)
SYNCP
TRIGINT(999999999)
```

Figure 16. Sortie standard d'une commande DISPLAY QMGR

Remarque : SYNCP est un attribut de gestionnaire de files d'attente en lecture seule.

Le paramètre **ALL** est la valeur par défaut de la commande **DISPLAY QMGR** . Il affiche tous les attributs du gestionnaire de files d'attente. En particulier, la sortie vous indique le nom du gestionnaire de files d'attente par défaut, le nom de la file d'attente de rebut et le nom de la file d'attente de commandes.

Vous pouvez confirmer que ces files d'attente existent en entrant la commande suivante:

```
DISPLAY QUEUE (SYSTEM.*)
```

La liste des files d'attente correspondant au radical `SYSTEM.*` s'affiche. Les parenthèses sont obligatoires.

Modification des attributs du gestionnaire de files d'attente

Pour modifier les attributs du gestionnaire de files d'attente spécifié dans la commande `runmqsc` , utilisez la commande `MQSC ALTER QMGR` en spécifiant les attributs et les valeurs que vous souhaitez modifier. Par exemple, utilisez les commandes suivantes pour modifier les attributs de `jupiter.queue.manager`:

```
runmqsc jupiter.queue.manager
ALTER QMGR DEADQ (ANOTHERDLQ) INHIBTEV (ENABLED)
```

La commande **ALTER QMGR** modifie la file d'attente de rebut utilisée et active les événements d'interdiction.

Information associée

[Création et gestion des gestionnaires de files d'attente sur Multiplatforms](#)

[Attributs du gestionnaire de files d'attente](#)

[runmqsc \(exécution des commandes MQSC\)](#)

[DISPLAY QMGR](#)

[ALTER QMGR](#)

Utilisation des files d'attente locales

Cette section contient des exemples de certaines commandes MQSC que vous pouvez utiliser pour gérer des files d'attente locales, modèles et alias.

Pour plus d'informations sur ces commandes, voir [Commandes MQSC](#) .

Information associée

[Restrictions de dénomination pour les files d'attente](#)

[Restrictions de dénomination pour d'autres objets](#)

Définition d'une file d'attente locale

Pour une application, le gestionnaire de files d'attente local est le gestionnaire de files d'attente auquel l'application est connectée. Les files d'attente gérées par le gestionnaire de files d'attente local sont dites locales pour ce gestionnaire de files d'attente.

Utilisez la commande MQSC **DEFINE QLOCAL** pour créer une file d'attente locale. Vous pouvez également utiliser la valeur par défaut définie dans la définition de file d'attente locale par défaut ou modifier les caractéristiques de la file d'attente locale par défaut.

Remarque : La file d'attente locale par défaut est nommée `SYSTEM.DEFAULT.LOCAL.QUEUE` et a été créée sur l'installation du système.

Par exemple, la commande **DEFINE QLOCAL** qui suit définit une file d'attente appelée `ORANGE.LOCAL.QUEUE` avec les caractéristiques suivantes:

- Il est activé pour les requêtes, activé pour les insertions et fonctionne en fonction de l'ordre de priorité.

- Il s'agit d'une file d'attente *normale* ; il ne s'agit pas d'une file d'attente d'initialisation ou de transmission et elle ne génère pas de messages de déclenchement.
- La longueur maximale de la file d'attente est de 5000 messages ; la longueur maximale est de 4194304 octets.

```
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) +
DESCR('Queue for messages from other systems') +
PUT(ENABLED) +
GET(ENABLED) +
NOTRIGGER +
MSGDLVSQ(PRIORITY) +
MAXDEPTH(5000) +
MAXMSGL(4194304) +
USAGE(NORMAL)
```

Remarque :

1. A l'exception de la valeur de la description, toutes les valeurs d'attribut affichées dans l'exemple sont les valeurs par défaut. Ces exemples sont inclus à des fins d'illustration. Vous pouvez les omettre si vous êtes sûr que les valeurs par défaut correspondent à celles que vous souhaitez ou qu'elles n'ont pas été modifiées. Voir aussi «[Affichage des attributs d'objet par défaut](#)», à la page 151.
2. **USAGE (NORMAL)** indique que cette file d'attente n'est pas une file d'attente de transmission.
3. Si vous disposez déjà d'une file d'attente locale sur le même gestionnaire de files d'attente avec le nom ORANGE.LOCAL.QUEUE, cette commande échoue. Utilisez l'attribut **REPLACE** si vous souhaitez remplacer la définition existante d'une file d'attente, mais voir aussi «[Modification des attributs de file d'attente locale](#)», à la page 152.

Information associée

[DEFINE QLOCAL](#)

Affichage des attributs d'objet par défaut

Vous pouvez utiliser la commande **DISPLAY QUEUE** pour afficher les attributs qui ont été extraits de l'objet par défaut lorsqu'un objet IBM MQ a été défini.

Lorsque vous définissez un objet IBM MQ, il prend tous les attributs que vous ne spécifiez pas à partir de l'objet par défaut. Par exemple, lorsque vous définissez une file d'attente locale, celle-ci hérite de tous les attributs que vous omettez dans la définition de la file d'attente locale par défaut, appelée SYSTEM.DEFAULT.LOCAL.QUEUE. Pour voir exactement quels sont ces attributs, utilisez la commande suivante:

```
DISPLAY QUEUE (SYSTEM.DEFAULT.LOCAL.QUEUE)
```

La syntaxe de cette commande est différente de celle de la commande **DEFINE** correspondante. Dans la commande **DISPLAY**, vous pouvez donner uniquement le nom de la file d'attente, tandis que dans la commande **DEFINE**, vous devez spécifier le type de la file d'attente, c'est-à-dire QLOCAL, QALIAS, QMODEL ou QREMOTE.

Vous pouvez afficher les attributs de manière sélective en les spécifiant individuellement. Exemple :

```
DISPLAY QUEUE (ORANGE.LOCAL.QUEUE) +
MAXDEPTH +
MAXMSGL +
CURDEPTH;
```

Cette commande affiche les trois attributs spécifiés comme suit:

```
AMQ8409: Display Queue details.
QUEUE(ORANGE.LOCAL.QUEUE)      TYPE(QLOCAL)
CURDEPTH(0)                     MAXDEPTH(5000)
MAXMSGL(4194304)
```

CURDEPTH est la longueur de la file d'attente en cours, c'est-à-dire le nombre de messages dans la file d'attente. Il s'agit d'un attribut utile à afficher, car en surveillant la longueur de la file d'attente, vous pouvez vous assurer que la file d'attente n'est pas saturée.

Information associée

[DISPLAY QUEUE](#)
[DEFINE QUEUES](#)

Copie d'une définition de file d'attente locale

Vous pouvez copier une définition de file d'attente à l'aide de l'attribut **LIKE** de la commande **DEFINE**.

Exemple :

```
DEFINE QLOCAL (MAGENTA.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE)
```

Cette commande crée une file d'attente avec les mêmes attributs que notre file d'attente d'origine ORANGE.LOCAL.QUEUE, plutôt que celles de la file d'attente locale par défaut du système. Entrez le nom de la file d'attente à copier exactement tel qu'il a été entré lors de la création de la file d'attente. Si le nom contient des caractères minuscules, placez-le entre apostrophes.

Vous pouvez également utiliser cette forme de la commande **DEFINE** pour copier une définition de file d'attente, mais remplacer une ou plusieurs modifications apportées aux attributs de l'original. Exemple :

```
DEFINE QLOCAL (THIRD.QUEUE) +  
LIKE (ORANGE.LOCAL.QUEUE) +  
MAXMSGL(1024);
```

Cette commande copie les attributs de la file d'attente ORANGE.LOCAL.QUEUE dans la file d'attente THIRD.QUEUE, mais spécifie que la longueur maximale des messages dans la nouvelle file d'attente doit être de 1024 octets au lieu de 4194304.

Remarque :

1. Lorsque vous utilisez l'attribut **LIKE** dans une commande **DEFINE**, vous copiez uniquement les attributs de file d'attente. Vous ne copiez pas les messages de la file d'attente.
2. Si vous définissez une file d'attente locale, sans spécifier **LIKE**, elle est identique à **DEFINE LIKE (SYSTEM.DEFAULT.LOCAL.QUEUE)**.

Information associée

[DEFINE QUEUES](#)

Modification des attributs de file d'attente locale

Vous pouvez modifier les attributs de file d'attente de deux manières, à l'aide de la commande **ALTER QLOCAL** ou de la commande **DEFINE QLOCAL** avec l'attribut **REPLACE**.

Dans «Définition d'une file d'attente locale», à la page 150, la file d'attente appelée ORANGE.LOCAL.QUEUE a été définie. Supposons, par exemple, que vous souhaitez réduire la longueur maximale des messages de cette file d'attente à 10 000 octets.

- A l'aide de la commande **ALTER** :

```
ALTER QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000)
```

Cette commande modifie un attribut unique, celui de la longueur maximale de message ; tous les autres attributs restent identiques.

- Utilisation de la commande **DEFINE** avec l'option **REPLACE** , par exemple:

```
DEFINE QLOCAL (ORANGE.LOCAL.QUEUE) MAXMSGL(10000) REPLACE
```

Cette commande modifie non seulement la longueur maximale des messages, mais également tous les autres attributs auxquels sont affectées leurs valeurs par défaut. La file d'attente est maintenant mise en file d'attente activée alors qu'auparavant elle était mise en file d'attente interdite. L'insertion activée est la valeur par défaut, comme indiqué par la file d'attente SYSTEM.DEFAULT.LOCAL.QUEUE.

Si vous réduisez la longueur maximale des messages dans une file d'attente existante, les messages existants ne sont pas affectés. Toutefois, tout nouveau message doit répondre aux nouveaux critères.

Information associée

[Files d'attente ALTER](#)

[ALTER QLOCAL](#)

[DEFINE QUEUES](#)

[DEFINE QLOCAL](#)

Effacement d'une file d'attente locale

Vous pouvez utiliser la commande **CLEAR** pour effacer une file d'attente locale.

Par exemple, pour supprimer tous les messages d'une file d'attente locale appelée MAGENTA.QUEUE, utilisez la commande suivante:

```
CLEAR QLOCAL (MAGENTA.QUEUE)
```

Remarque : Aucune invite ne vous permet de changer d'avis. Une fois que vous appuyez sur la touche Entrée, les messages sont perdus.

Vous ne pouvez pas effacer une file d'attente si:

- Des messages non validés ont été placés dans la file d'attente sous le point de synchronisation.
- La file d'attente est ouverte par une application.

Information associée

[QLOCAL CLEAR](#)

Suppression d'une file d'attente locale

Vous pouvez utiliser la commande MQSC **DELETE QLOCAL** pour supprimer une file d'attente locale.

Une file d'attente ne peut pas être supprimée si elle contient des messages non validés. Toutefois, si la file d'attente comporte un ou plusieurs messages validés et aucun message non validé, elle ne peut être supprimée que si vous spécifiez l'option **PURGE** . Exemple :

```
DELETE QLOCAL (PINK.QUEUE) PURGE
```

La spécification de **NOPURGE** à la place de **PURGE** garantit que la file d'attente n'est pas supprimée si elle contient des messages validés.

Information associée

[SUPPRIMER QLOCAL](#)

Exploration des files d'attente

IBM MQ fournit un exemple de navigateur de files d'attente que vous pouvez utiliser pour examiner le contenu des messages d'une file d'attente. Le navigateur est fourni dans les formats source et exécutable.

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

Windows Sous Windows, les noms de fichier et les chemins d'accès de l'exemple de navigateur de files d'attente sont les suivants:

Source

`MQ_INSTALLATION_PATH\tools\c\samples\`

Exécutable

`MQ_INSTALLATION_PATH\tools\c\samples\bin\amqsbcg.exe`

Linux **UNIX** Sous UNIX et Linux, les noms de fichier et les chemins d'accès sont les suivants:

Source

`MQ_INSTALLATION_PATH/samp/amqsbcg0.c`

Exécutable

`MQ_INSTALLATION_PATH/samp/bin/amqsbcg`

L'exemple requiert deux paramètres d'entrée, le nom de la file d'attente et le nom du gestionnaire de files d'attente. Exemple :

```
amqsbcg SYSTEM.ADMIN.QMGREVENT.tpp01 saturn.queue.manager
```

Les résultats typiques de cette commande sont présentés dans [Figure 17, à la page 155](#).

Certains utilitaires, tels que tar, ne peuvent pas gérer les fichiers de plus de 2 Go. Avant d'activer la prise en charge des fichiers volumineux, consultez la documentation de votre système d'exploitation pour plus d'informations sur les restrictions relatives aux utilitaires que vous utilisez.

Pour plus d'informations sur la planification de la quantité de stockage dont vous avez besoin pour les files d'attente, voir [IBM MQ Family-Performance Reports](#) pour les rapports de performances spécifiques à la plateforme.

Utilisation de files d'attente alias

Vous pouvez définir une file d'attente alias pour faire référence indirectement à une autre file d'attente ou rubrique.

V 9.0.1 > V 9.0.0.1



Avertissement : Les listes de distribution ne prennent pas en charge les files d'attente alias qui pointent vers des objets de rubrique. Depuis IBM MQ 9.0.1 et IBM MQ 9.0.0 Fix Pack 1, si une file d'attente alias pointe vers un objet de rubrique dans une liste de distribution, IBM MQ renvoie MQRC_ALIAS_BASE_Q_TYPE_ERROR.

La file d'attente à laquelle une file d'attente alias fait référence peut être l'une des suivantes:

- Une file d'attente locale (voir «[Définition d'une file d'attente locale](#)», à la page 150).
- Définition locale d'une file d'attente éloignée (voir «[Création d'une définition locale d'une file d'attente éloignée](#)», à la page 207).
- Rubrique.

Une file d'attente alias n'est pas une file d'attente réelle, mais une définition qui se résout en file d'attente réelle (ou cible) lors de l'exécution. La définition de file d'attente alias indique la file d'attente cible. Lorsqu'une application effectue un appel MQOPEN vers une file d'attente alias, le gestionnaire de files d'attente résout l'alias en nom de file d'attente cible.

Une file d'attente alias ne peut pas être résolue en une autre file d'attente alias définie localement. Toutefois, une file d'attente alias peut être convertie en files d'attente alias définies ailleurs dans des clusters dont le gestionnaire de files d'attente local est membre. Pour plus d'informations, voir [Résolution de nom](#) .

Les files d'attente alias sont utiles pour:

- Attribution à différentes applications de différents niveaux de droits d'accès à la file d'attente cible.
- Permettre à différentes applications de travailler avec la même file d'attente de différentes manières. (Vous souhaitez peut-être affecter des priorités par défaut différentes ou des valeurs de persistance par défaut différentes.)
- Simplification de la maintenance, de la migration et de l'équilibrage de la charge de travail. (Vous souhaitez peut-être modifier le nom de la file d'attente cible sans avoir à modifier votre application, qui continue d'utiliser l'alias.)

Par exemple, supposons qu'une application ait été développée pour placer des messages dans une file d'attente appelée MY.ALIAS.QUEUE. Il indique le nom de cette file d'attente lorsqu'il émet une demande MQOPEN et, indirectement, s'il insère un message dans cette file d'attente. L'application ne sait pas que la file d'attente est une file d'attente d'alias. Pour chaque appel MQI utilisant cet alias, le gestionnaire de files d'attente résout le nom de la file d'attente réelle, qui peut être une file d'attente locale ou une file d'attente éloignée définie sur ce gestionnaire de files d'attente.

En modifiant la valeur de l'attribut TARGQ, vous pouvez rediriger les appels MQI vers une autre file d'attente, éventuellement sur un autre gestionnaire de files d'attente. Cela est utile pour la maintenance, la migration et l'équilibrage de charge.

Définition d'une file d'attente alias

La commande suivante crée une file d'attente alias:

```
DEFINE QALIAS (MY.ALIAS.QUEUE) TARGET (YELLOW.QUEUE)
```

Cette commande redirige les appels MQI qui spécifient MY.ALIAS.QUEUE dans la file d'attente YELLOW.QUEUE. La commande ne crée pas la file d'attente cible ; les appels MQI échouent si la file d'attente est YELLOW.QUEUE n'existe pas lors de l'exécution.

Si vous modifiez la définition d'alias, vous pouvez rediriger les appels MQI vers une autre file d'attente. Exemple :

```
ALTER QALIAS (MY.ALIAS.QUEUE) TARGET (MAGENTA.QUEUE)
```

Cette commande redirige les appels MQI vers une autre file d'attente, MAGENTA.QUEUE.

Vous pouvez également utiliser des files d'attente alias pour qu'une seule file d'attente (la file d'attente cible) ait des attributs différents pour différentes applications. Pour ce faire, vous définissez deux alias, un pour chaque application. Supposons qu'il existe deux applications:

- L'application ALPHA peut placer des messages sur YELLOW.QUEUE, mais n'est pas autorisé à en extraire des messages.
- L'application bêta peut recevoir des messages de YELLOW.QUEUE, mais n'est pas autorisé à y placer des messages.

La commande suivante définit un alias qui est activé et désactivé pour l'application ALPHA:

```
DEFINE QALIAS (ALPHAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (ENABLED) +  
GET (DISABLED)
```

La commande suivante définit un alias qui est mis à l'état désactivé et activé pour l'application BETA:

```
DEFINE QALIAS (BETAS.ALIAS.QUEUE) +  
TARGET (YELLOW.QUEUE) +  
PUT (DISABLED) +  
GET (ENABLED)
```

ALPHA utilise le nom de file d'attente ALPHAS.ALIAS.QUEUE dans ses appels MQI ; BETA utilise le nom de file d'attente BETAS.ALIAS.QUEUE. Ils accèdent tous deux à la même file d'attente, mais de différentes manières.

Vous pouvez utiliser les attributs LIKE et REPLACE lorsque vous définissez des alias de file d'attente, de la même manière que vous utilisez ces attributs avec des files d'attente locales.

Utilisation d'autres commandes avec des files d'attente alias

Vous pouvez utiliser les commandes MQSC appropriées pour afficher ou modifier les attributs de file d'attente alias ou pour supprimer l'objet file d'attente alias. Exemple :

Utilisez la commande **DISPLAY QALIAS** pour afficher les attributs de la file d'attente alias:

```
DISPLAY QUEUE (ALPHAS.ALIAS.QUEUE)
```

Utilisez la commande **ALTER QALIAS** pour modifier le nom de la file d'attente de base, dans laquelle l'alias est résolu, où l'option `force` force la modification même si la file d'attente est ouverte:

```
ALTER QALIAS (ALPHAS.ALIAS.QUEUE) TARGQ(ORANGE.LOCAL.QUEUE) FORCE
```

Utilisez la commande **DELETE QALIAS** pour supprimer cet alias de file d'attente:

```
DELETE QALIAS (ALPHAS.ALIAS.QUEUE)
```

Vous ne pouvez pas supprimer une file d'attente alias si la file d'attente est ouverte pour une application.

Information associée

[ALTER QALIAS](#)

[DEFINE QALIAS](#)

[DELETE QALIAS](#)

[Listes de diffusion](#)

Utilisation des files d'attente de rebut

Chaque gestionnaire de files d'attente possède généralement une file d'attente locale à utiliser comme file d'attente de rebut, de sorte que les messages qui ne peuvent pas être distribués à leur destination correcte puissent être stockés en vue d'une extraction ultérieure. Vous indiquez au gestionnaire de files d'attente la file d'attente de rebut et indiquez comment les messages trouvés dans une file d'attente de rebut doivent être traités. L'utilisation de files d'attente de rebut peut affecter la séquence dans laquelle les messages sont distribués. Vous pouvez donc choisir de ne pas les utiliser.

Pour indiquer au gestionnaire de files d'attente le nom de la file d'attente de rebut, indiquez un nom de file d'attente de rebut dans la commande **crtmqm** (`crtmqm -u DEAD.LETTER.QUEUE`, par exemple) ou utilisez l'attribut **DEADQ** dans la commande **ALTER QMGR** pour en spécifier une ultérieurement. Vous devez définir la file d'attente de rebut avant de l'utiliser.

Exemple de file d'attente de rebut appelée SYSTEM.DEAD.LETTER.QUEUE est disponible avec le produit. Cette file d'attente est automatiquement créée lorsque vous créez le gestionnaire de files d'attente. Vous pouvez modifier cette définition si nécessaire et la renommer.

Une file d'attente de rebut n'a pas d'exigences particulières si ce n'est que:

- Il doit s'agir d'une file d'attente locale
- Son attribut MAXMSGL (longueur maximale de message) doit permettre à la file d'attente de recevoir les messages les plus volumineux que le gestionnaire de files d'attente doit traiter **plus** la taille de l'en-tête de message non livrés (MQDLH)

L'utilisation de files d'attente de rebut peut affecter la séquence dans laquelle les messages sont distribués. Vous pouvez donc choisir de ne pas les utiliser. Vous définissez l'attribut de canal USEDQ pour déterminer si la file d'attente de rebut est utilisée lorsque les messages ne peuvent pas être distribués. Cet attribut peut être configuré de sorte que certaines fonctions du gestionnaire de files d'attente utilisent la file d'attente de rebut alors que d'autres fonctions ne l'utilisent pas. Pour plus d'informations sur l'utilisation de l'attribut de canal USEDQ avec différentes commandes MQSC, voir [DEFINE CHANNEL](#), [DISPLAY CHANNEL](#), [ALTER CHANNEL](#) et [DISPLAY CLUSQMGR](#).

IBM MQ fournit un gestionnaire de files d'attente de rebut qui vous permet de spécifier comment les messages trouvés dans une file d'attente de rebut doivent être traités ou supprimés. Voir [«Traitement des messages dans une file d'attente de rebut IBM MQ»](#), à la page 158.

Information associée

[Files d'attente de rebut](#)

[Traitement des incidents liés aux messages non distribués](#)

[ALTER QMGR](#)

[crtmqm \(création du gestionnaire de files d'attente\)](#)

Traitement des messages dans une file d'attente de rebut IBM MQ

Pour traiter les messages dans une file d'attente de messages non livrés, IBM MQ fournit un gestionnaire de files d'attente de messages non livrés par défaut. Le gestionnaire compare les messages de la file d'attente des messages non livrés aux entrées d'une table de règles que vous définissez.

Les messages peuvent être placés sur un DLQ par des gestionnaires de files d'attente, des agents MCA et des applications. Tous les messages de la file d'attente de rebut doivent être précédés d'une structure d'*en-tête de non distribution*, MQDLH. Les messages insérés dans la file d'attente des messages non livrés par un gestionnaire de files d'attente ou un agent MCA ont toujours cet en-tête ; les applications qui placent des messages dans la file d'attente des messages non livrés doivent fournir cet en-tête. La zone *Reason* de la structure MQDLH contient un code anomalie indiquant la raison pour laquelle le message se trouve dans la file d'attente de rebut.

Tous les environnements IBM MQ ont besoin d'une routine pour traiter régulièrement les messages sur le DLQ. IBM MQ fournit une routine par défaut, appelée *gestionnaire de files d'attente de rebut* (gestionnaire DLQ), que vous appelez à l'aide de la commande `runmqdlq`.

Les instructions de traitement des messages sur le DLQ sont fournies au gestionnaire DLQ au moyen d'une *table de règles* écrite par l'utilisateur. En d'autres termes, le gestionnaire DLQ met en correspondance les messages de la DLQ avec les entrées de la table de règles ; lorsqu'un message DLQ correspond à une entrée de la table de règles, le gestionnaire DLQ exécute l'action associée à cette entrée.

Information associée

[Files d'attente de rebut](#)

[Traitement des incidents liés aux messages non distribués](#)

Gestionnaire de file d'attente de rebut sous IBM i

Qu'est-ce que le gestionnaire de files d'attente de rebut IBM i et comment l'invoquer?

Une *file d'attente de messages non livrés* (DLQ), parfois appelée *file d'attente de messages non livrés*, est une file d'attente des messages qui ne peuvent pas être distribués à leurs files d'attente de destination. Chaque gestionnaire de files d'attente d'un réseau doit être associé à un DLQ.

Remarque : Il est souvent préférable d'éviter de placer des messages sur un DLQ. Pour plus d'informations sur l'utilisation et l'évitement des DLQ, voir [«Utilisation des files d'attente de rebut»](#), à la page 158.

Les gestionnaires de files d'attente, les agents MCA et les applications peuvent insérer des messages dans la file d'attente des messages non livrés. Tous les messages de la file d'attente de rebut doivent être précédés d'une structure d'*en-tête de non distribution*, MQDLH. Les messages insérés dans la file d'attente des messages non livrés par un gestionnaire de files d'attente ou par un agent de canal de transmission de messages ont toujours un MQDLH. Fournissez toujours un MQDLH aux applications qui placent des messages sur le DLQ. La zone *Reason* de la structure MQDLH contient un code anomalie indiquant la raison pour laquelle le message se trouve dans la file d'attente de rebut.

Dans tous les environnements IBM MQ, il doit y avoir une routine qui s'exécute régulièrement pour traiter les messages sur le DLQ. IBM MQ fournit une routine par défaut, appelée *gestionnaire de files d'attente de rebut* (gestionnaire DLQ), que vous appelez à l'aide de la commande `STRMQMDLQ`. Une *table de règles* écrite par l'utilisateur fournit des instructions au gestionnaire DLQ pour le traitement des messages sur le DLQ. En d'autres termes, le gestionnaire DLQ compare les messages de la file d'attente DLQ aux entrées de la table de règles. Lorsqu'un message DLQ correspond à une entrée de la table des règles, le gestionnaire DLQ effectue l'action associée à cette entrée.

Appel du gestionnaire DLQ

Utilisez la commande `STRMQMDLQ` pour appeler le gestionnaire DLQ. Vous pouvez nommer la file d'attente des messages non livrés à traiter et le gestionnaire de files d'attente à utiliser de deux manières:

- En tant que paramètres de `STRMQMDLQ` à partir de l'invite de commande. Exemple :

```
STRMQMDLQ UDLMSGQ(ABC1.DEAD.LETTER.QUEUE) SRCMBR(QRULE) SRCFILE(library/QXTSRC)
MQMNAME(MY.QUEUE.MANAGER)
```

- Dans la table de règles. Exemple :

INPUTQ(ABC1.DEAD.LETTER.QUEUE)

Remarque : La table de règles est un membre d'un fichier physique source qui peut prendre n'importe quel nom.

Les exemples s'appliquent à la file d'attente des messages non livrés appelée ABC1.DEAD.LETTER.QUEUE, détenue par le gestionnaire de files d'attente par défaut.

Si vous ne spécifiez pas le DLQ ou le gestionnaire de files d'attente comme indiqué, le gestionnaire de files d'attente par défaut pour l'installation est utilisé avec le DLQ appartenant à ce gestionnaire de files d'attente.

La commande STRMQMDLQ extrait son entrée de la table de règles.

Vous devez être autorisé à accéder à la file d'attente des messages non livrés et aux files d'attente de messages auxquelles les messages de la file d'attente des messages non livrés sont transférés afin d'exécuter le gestionnaire de file d'attente des messages non livrés. Vous devez également être autorisé à utiliser l'identité d'autres utilisateurs pour que le DLQ puisse placer des messages dans des files d'attente avec les droits de l'ID utilisateur dans le contexte de message.

Information associée

[Files d'attente de rebut](#)

[Traitement des incidents liés aux messages non distribués](#)

Table des règles du gestionnaire DLQ sous IBM i

La table des règles du gestionnaire de files d'attente de rebut définit comment le gestionnaire de files d'attente de rebut traite les messages qui arrivent sur la file d'attente de rebut IBM i.

La table des règles du gestionnaire DLQ définit comment le gestionnaire DLQ doit traiter les messages qui arrivent sur le DLQ. Il existe deux types d'entrée dans une table de règles:

- La première entrée de la table, qui est facultative, contient des *données de contrôle*.
- Toutes les autres entrées de la table sont des *règles* que le gestionnaire DLQ doit suivre. Chaque règle se compose d'un *modèle* (un ensemble de caractéristiques de message) auquel un message est comparé et d'une *action* à effectuer lorsqu'un message sur la file d'attente des messages non livrés correspond au modèle spécifié. Une table de règles doit contenir au moins une règle.

Chaque entrée de la table de règles comprend un ou plusieurs mots clés.

Données de contrôle

Cette section décrit les mots clés que vous pouvez inclure dans une entrée de données de contrôle dans une table de règles de gestionnaire DLQ. Notez ce qui suit :

- La valeur par défaut d'un mot clé, le cas échéant, est soulignée.
- La ligne verticale (|) sépare les alternatives. Vous ne pouvez spécifier qu'une seule de ces valeurs.
- Tous les mots clés sont facultatifs.

INPUTQ (*QueueName* | ')

Nom de la file d'attente des données à traiter:

1. Toute valeur UDLMMSGQ (ou *DFT) que vous indiquez comme paramètre de la commande **STRMQMDLQ** remplace toute valeur INPUTQ dans la table de règles.
2. Si vous indiquez une valeur UDLMMSGQ vide comme paramètre de la commande **STRMQMDLQ**, la valeur INPUTQ de la table de règles est utilisée.
3. Si vous indiquez une valeur UDLMMSGQ à blanc comme paramètre de la commande **STRMQMDLQ** et une valeur INPUTQ à blanc dans la table de règles, la file d'attente de rebut par défaut du système est utilisée.

INPUTQM (*QueueManagerNom* | ' ')

Nom du gestionnaire de files d'attente propriétaire de la file d'attente des messages non livrés indiquée dans le mot clé INPUTQ.

Si vous ne spécifiez pas de gestionnaire de files d'attente ou que vous spécifiez INPUTQM ('') dans la table de règles, le système utilise le gestionnaire de files d'attente par défaut pour l'installation.

RETRYINT (*Intervalle* | 60)

Intervalle, en secondes, auquel le gestionnaire DLQ doit tenter de retraiter les messages sur le DLQ qui n'ont pas pu être traités lors de la première tentative et pour lesquels des tentatives répétées ont été demandées. Par défaut, l'intervalle entre les nouvelles tentatives est de 60 secondes.

WAIT (YES|NO|*nnn*)

Indique si le gestionnaire DLQ doit attendre que d'autres messages arrivent sur le DLQ lorsqu'il détecte qu'il ne peut pas traiter d'autres messages.

YES

Provoque l'attente indéfinie du gestionnaire de la file d'attente des messages non livrés.

NO

Provoque l'arrêt du gestionnaire DLQ lorsqu'il détecte que le DLQ est vide ou qu'il ne contient aucun message qu'il peut traiter.

nnn

Fait en sorte que le gestionnaire de la file d'attente des messages non livrés attende *nnn* secondes pour que le nouveau travail arrive avant de s'arrêter, après avoir détecté que la file d'attente est vide ou qu'elle ne contient aucun message pouvant être traité.

Indiquez WAIT (YES) pour les DLQ occupées et WAIT (NO) ou WAIT (*nnn*) pour les DLQ dont le niveau d'activité est faible. Si le gestionnaire DLQ est autorisé à s'arrêter, appelez-le à nouveau à l'aide du déclenchement.

Vous pouvez indiquer le nom du DLQ comme paramètre d'entrée de la commande **STRMQMDLQ**, au lieu d'inclure des données de contrôle dans la table de règles. Si une valeur est spécifiée à la fois dans la table de règles et en entrée de la commande **STRMQMDLQ**, la valeur spécifiée dans la commande **STRMQMDLQ** est prioritaire.

Remarque : Si une entrée de données de contrôle est incluse dans la table de règles, elle doit être la première entrée de la table.

Règles DLQ (modèles et actions) sur IBM i

Description des modèles et des actions pour chacune des règles de file d'attente de rebut IBM i.

Voici un exemple de règle d'une table de règles de gestionnaire DLQ:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +  
ACTION (RETRY) RETRY (3)
```

Cette règle demande au gestionnaire DLQ d'effectuer 3 tentatives de distribution à sa file d'attente de destination de tout message persistant qui a été inséré dans la file d'attente DLQ car MQPUT et MQPUT1 ont été inhibés.

Cette section décrit les mots clés que vous pouvez inclure dans une règle. Notez ce qui suit :

- La valeur par défaut d'un mot clé, le cas échéant, est soulignée. Pour la plupart des mots clés, la valeur par défaut est * (astérisque), qui correspond à n'importe quelle valeur.
- La ligne verticale (|) sépare les alternatives. Vous ne pouvez spécifier qu'une seule de ces valeurs.
- Tous les mots clés sauf ACTION sont facultatifs.

Cette section commence par une description des mots clés de correspondance de modèle (ceux avec lesquels les messages du DLQ sont mis en correspondance). Il décrit ensuite les mots clés d'action (ceux qui déterminent comment le gestionnaire DLQ doit traiter un message correspondant).

Les mots clés de correspondance de modèle sont décrits dans un exemple. Utilisez ces mots clés pour spécifier les valeurs par rapport auxquelles les messages de la file d'attente de rebut IBM i sont mis en correspondance. Tous les mots clés de correspondance de modèle sont facultatifs.

APPLIDAT (*DonnéesApplIdentity*|*)

Valeur *ApplIdentityData* du message sur la file d'attente des messages non livrés, indiquée dans le descripteur de message, MQMD.

APPLNAME (*PutApplNom*|*)

Nom de l'application qui a émis l'appel MQPUT ou MQPUT1 , comme indiqué dans la zone *PutApplName* du descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

APPLTYPE (*PutApplType*|*)

Valeur *PutApplType* spécifiée dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

DESTQ (*QueueName*|*)

Nom de la file d'attente de messages à laquelle le message est destiné.

DESTQM (*QueueManagerNom*|*)

Nom du gestionnaire de files d'attente de la file d'attente de messages à laquelle le message est destiné.

FEEDBACK (*Commentaires en retour*|*)

Lorsque la valeur *MsgType* est MQMT_REPORT, *Feedback* décrit la nature du rapport.

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQFB_COA pour identifier les messages de la file d'attente des messages non livrés qui nécessitent une confirmation de leur arrivée dans leurs files d'attente de destination.

FORMAT (*Format*|*)

Nom utilisé par l'expéditeur du message pour décrire le format des données du message.

MSGTYPE (*MsgType*|*)

Type du message sur la file d'attente des messages non livrés.

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQMT_REQUEST pour identifier les messages sur le DLQ qui nécessitent des réponses.

PERSISTANT (*Persistence*|*)

Valeur de persistance du message. (La persistance d'un message détermine s'il survit aux redémarrages du gestionnaire de files d'attente.)

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQPER_PERSISTENT pour identifier les messages sur le DLQ qui sont persistants.

REASON (*ReasonCode*|*)

Code raison décrivant la raison pour laquelle le message a été inséré dans le DLQ.

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQRC_Q_FULL pour identifier les messages placés dans la file d'attente des messages non livrés car leurs files d'attente de destination étaient saturées.

REPLYQ (*QueueName*|*)

Nom de la file d'attente de réponse indiqué dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

REPLYQM (*QueueManagerNom*|*)

Nom du gestionnaire de files d'attente de la file d'attente de réponses spécifiée dans le mot clé REPLYQ.

USERID (*UserIdentifier* | *)

ID de l'utilisateur à l'origine du message sur la file d'attente des messages non livrés, comme indiqué dans le descripteur de message, MQMD.

IBM i**Mots clés d'action DLQ sous IBM i**

Utilisez ces mots clés d'action de file d'attente de rebut pour déterminer comment un message correspondant dans la file d'attente de rebut IBM i est traité.

ACTION (DISCARD | IGNORE | RETRY | FWD)

Action effectuée pour tout message de la file d'attente des messages non livrés qui correspond au modèle défini dans cette règle.

SUPPRIMER

Entraîne la suppression du message de la file d'attente des messages non livrés.

IGNORE

Permet de conserver le message dans la file d'attente des messages non livrés.

REESSAYER

Permet au gestionnaire de file d'attente des messages non livrés de tenter à nouveau d'insérer le message dans sa file d'attente de destination.

Transfert

Permet au gestionnaire DLQ de transmettre le message à la file d'attente indiquée dans le mot clé FWDQ.

Vous devez spécifier le mot clé ACTION. Le nombre de tentatives d'implémentation d'une action est régi par le mot clé RETRY. Le mot clé RETRYINT des données de contrôle contrôle l'intervalle entre les tentatives.

FWDQ (*QueueName* | & DESTQ | & REPLYQ)

Nom de la file d'attente de messages vers laquelle le message est réacheminé lorsque vous sélectionnez le mot clé ACTION.

QueueName

Nom d'une file d'attente de messages. FWDQ (") n'est pas valide.

& DESTQ

Prenez le nom de la file d'attente dans la zone *DestQName* de la structure MQDLH.

& REPLYQ

Prenez le nom de la file d'attente dans la zone *ReplyToQ* du descripteur de message, MQMD.

Vous pouvez spécifier REPLYQ (? *) dans le modèle de message pour éviter les messages d'erreur, lorsqu'une règle spécifiant FWDQ (& REPLYQ) correspond à un message dont la zone *ReplyToQ* est vide.

FWDQM (*QueueManagerNom* | & DESTQM | & REPLYQM | ')

Gestionnaire de files d'attente de la file d'attente vers laquelle un message est réacheminé.

QueueManagerName

Nom du gestionnaire de files d'attente de la file d'attente vers laquelle le message est réacheminé lorsque vous sélectionnez le mot clé ACTION (FWD).

& DESTQM

Utilisez le nom du gestionnaire de files d'attente dans la zone *DestQMGrNom* de la structure MQDLH.

& REPLYQM

Utilisez le nom du gestionnaire de files d'attente dans la zone *ReplyToQMGr* du descripteur de message, MQMD.

..

FWDQM ("), qui est la valeur par défaut, identifie le gestionnaire de files d'attente local.

HEADER (OUI|NON)

Indique si le MQDLH doit rester sur un message pour lequel ACTION (FWD) est demandée. Par défaut, le MQDLH reste sur le message. Le mot clé HEADER n'est pas valide pour les actions autres que FWD.

PUTAUT (DEF| CTX)

Droits avec lesquels les messages doivent être insérés par le gestionnaire DLQ:

DEF

Insère des messages avec les droits du gestionnaire DLQ lui-même.

CTX

Permet d'insérer les messages avec les droits de l'ID utilisateur dans le contexte du message.

Vous devez être autorisé à assumer l'identité d'autres utilisateurs, si vous indiquez PUTAUT (CTX).

RETRY (*Nombre de nouvelles tentatives*|1)

Nombre de fois, compris entre 1 et 999 999 999, où une action doit être tentée (à l'intervalle indiqué dans le mot clé RETRYINT des données de contrôle).

Remarque : Le nombre de tentatives effectuées par le gestionnaire de file d'attente des messages non livrés pour implémenter une règle particulière est spécifique à l'instance en cours du gestionnaire de file d'attente des messages non livrés ; ce nombre n'est pas conservé lors des redémarrages. Si vous redémarrez le gestionnaire DLQ, le nombre de tentatives d'application d'une règle est réinitialisé à zéro.

IBM i Conventions de la table de règles DLQ sous IBM i

La table de règles de file d'attente de rebut IBM i doit respecter des conventions spécifiques concernant sa syntaxe, sa structure et son contenu.

- Une table de règles doit contenir au moins une règle.
- Les mots clés peuvent apparaître dans n'importe quel ordre.
- Un mot clé ne peut être inclus qu'une seule fois dans une règle.
- Les mots clés ne sont pas sensibles à la casse.
- Un mot clé et sa valeur de paramètre doivent être séparés des autres mots clés par au moins un espace ou une virgule.
- N'importe quel nombre de blancs peut apparaître au début ou à la fin d'une règle et entre les mots clés, la ponctuation et les valeurs.
- Chaque règle doit commencer sur une nouvelle ligne.
- Pour la portabilité, la longueur significative d'une ligne ne doit pas dépasser 72 caractères.
- Utilisez le signe plus (+) comme dernier caractère non blanc sur une ligne pour indiquer que la règle se poursuit à partir du premier caractère non blanc de la ligne suivante. Utilisez le signe moins (-) comme dernier caractère non blanc sur une ligne pour indiquer que la règle se poursuit à partir du début de la ligne suivante. Les caractères de continuation peuvent apparaître dans les mots clés et les paramètres.

Exemple :

```
APPLNAME (' ABC+  
D')
```

gène're'ABCD'.

```
APPLNAME (' ABC-  
D')
```

gène're'ABC D'.

- Les lignes de commentaire, qui commencent par un astérisque (*), peuvent apparaître n'importe où dans la table de règles.
- Les lignes vides sont ignorées.

- Chaque entrée de la table de règles du gestionnaire DLQ comprend un ou plusieurs mots clés et leurs paramètres associés. Les paramètres doivent respecter les règles de syntaxe suivantes:
 - Chaque valeur de paramètre doit comporter au moins un caractère significatif. Les guillemets de délimitation dans les valeurs placées entre guillemets ne sont pas considérés comme significatifs. Par exemple, les paramètres suivants sont valides:

| | |
|--------------------|----------------------------|
| FORMAT (' ABC ') | 3 caractères significatifs |
| FORMAT (ABC) | 3 caractères significatifs |
| FORMAT (' A ') | 1 caractère significatif |
| FORMAT (A) | 1 caractère significatif |
| FORMAT (' ') | 1 caractère significatif |

Ces paramètres ne sont pas valides car ils ne contiennent pas de caractères significatifs:

```
FORMAT ( ' ' )
FORMAT ( )
FORMAT ( )
FORMAT
```

- Les caractères génériques sont pris en charge. Vous pouvez utiliser le point d'interrogation (?) à la place d'un caractère unique, à l'exception d'un blanc de fin. Vous pouvez utiliser l'astérisque (*) à la place de zéro ou plusieurs caractères adjacents. L'astérisque (*) et le point d'interrogation (?) sont **toujours** interprétés comme des caractères génériques dans les valeurs de paramètre.
- Vous ne pouvez pas inclure de caractères génériques dans les paramètres des mots clés suivants: ACTION, HEADER, RETRY, FWDQ, FWDQM et PUTAUT.
- Les blancs de fin dans les valeurs de paramètre et dans les zones correspondantes du message sur le DLQ ne sont pas importants lors de l'exécution de correspondances de caractères génériques. Toutefois, les blancs de début et les blancs imbriqués dans les chaînes entre guillemets sont importants pour les correspondances de caractères génériques.
- Les paramètres numériques ne peuvent pas inclure le caractère générique point d'interrogation (?). Vous pouvez inclure l'astérisque (*) à la place d'un paramètre numérique entier, mais l'astérisque ne peut pas être inclus dans un paramètre numérique. Par exemple, les paramètres numériques suivants sont valides:

| | |
|-------------------|--|
| MSGTYPE (2) | Seuls les messages de réponse sont éligibles |
| MSGTYPE (*) | Tout type de message est éligible |
| MSGTYPE (' * ') | Tout type de message est éligible |

Toutefois, MSGTYPE (' 2* ') n'est pas valide car il inclut un astérisque (*) dans un paramètre numérique.

- Les paramètres numériques doivent être compris entre 0 et 999 999 999. Si la valeur de paramètre est comprise dans cette plage, elle est acceptée, même si elle n'est pas actuellement valide dans la zone à laquelle le mot clé est associé. Vous pouvez utiliser des noms symboliques pour les paramètres numériques.
- Si une valeur de chaîne est plus courte que la zone de la MQDLH ou de la MQMD à laquelle le mot clé se rapporte, la valeur est remplie avec des blancs à la longueur de la zone. Si la valeur, à l'exclusion des astérisques, est plus longue que la zone, une erreur est diagnostiquée. Par exemple, voici toutes les valeurs de chaîne valides pour une zone de 8 caractères:

| | |
|---------------|--|
| ' ABCDEFGH ' | 8 caractères |
| ' A*C*E*G*I ' | 5 caractères à l'exclusion des astérisques |

'*A*C*E*G*I*K*M*O*'

8 caractères à l'exclusion des astérisques

- Les chaînes qui contiennent des blancs, des caractères minuscules ou des caractères spéciaux autres que le point (.), la barre oblique (?), le trait de soulignement (_) et le signe de pourcentage (%) doivent être placées entre apostrophes. Les caractères minuscules qui ne sont pas placés entre guillemets sont convertis en majuscules. Si la chaîne comprend un guillemet, deux apostrophes doivent être utilisées pour indiquer à la fois le début et la fin du guillemet. Lorsque la longueur de la chaîne est calculée, chaque occurrence de guillemets est comptée comme un caractère unique.

IBM i *Traitement de la table de règles DLQ sous IBM i*

Le gestionnaire de files d'attente de rebut recherche dans la table de règles une règle dont le modèle correspond à un message de la file d'attente de rebut IBM i .

La recherche commence par la première règle de la table et se poursuit séquentiellement dans la table. Lorsqu'une règle avec un modèle de correspondance est trouvée, la table de règles tente l'action à partir de cette règle. Le gestionnaire DLQ incrémente le nombre de nouvelles tentatives pour une règle de 1 chaque fois qu'il tente d'appliquer cette règle. Si la première tentative échoue, la tentative est répétée jusqu'à ce que le nombre de tentatives effectuées corresponde au nombre indiqué dans le mot clé RETRY. Si toutes les tentatives échouent, le gestionnaire DLQ recherche la règle de correspondance suivante dans la table.

Ce processus est répété pour les règles de correspondance suivantes jusqu'à ce qu'une action aboutisse. Lorsque chaque règle correspondante a été tentée le nombre de fois indiqué dans son mot clé RETRY et que toutes les tentatives ont échoué, ACTION (IGNORE) est utilisé. ACTION (IGNORE) est également pris en compte si aucune règle de correspondance n'est trouvée.

Remarque :

1. Les modèles de règle de correspondance sont recherchés uniquement pour les messages de la file d'attente des messages non livrés qui commencent par un MQDLH. Les messages qui ne commencent pas par un MQDLH sont signalés périodiquement comme étant erronés et restent indéfiniment sur le DLQ.
2. Tous les mots clés de modèle peuvent être définis par défaut, de sorte qu'une règle ne peut être constituée que d'une action. Notez toutefois que les règles d'action uniquement sont appliquées à tous les messages de la file d'attente qui ont des MQDLH et qui n'ont pas déjà été traités conformément aux autres règles de la table.
3. La table de règles est validée lorsque le gestionnaire DLQ démarre et les erreurs sont signalées à ce moment-là. (Les messages d'erreur émis par le gestionnaire DLQ sont décrits dans [Messages et codes anomalie](#).) Vous pouvez apporter des modifications à la table de règles à tout moment, mais ces modifications n'entrent en vigueur que lorsque le gestionnaire DLQ est redémarré.
4. Le gestionnaire DLQ ne modifie pas le contenu des messages, du MQDLH ou du descripteur de message. Le gestionnaire DLQ insère toujours des messages dans d'autres files d'attente avec l'option de message MQPMO_PASS_ALL_CONTEXT.
5. Les erreurs de syntaxe consécutives dans la table de règles peuvent ne pas être reconnues, car la validation de la table de règles élimine la génération d'erreurs répétitives.
6. Le gestionnaire DLQ ouvre le DLQ avec l'option MQOO_INPUT_AS_Q_DEF.
7. Plusieurs instances du gestionnaire DLQ peuvent s'exécuter simultanément sur la même file d'attente, à l'aide de la même table de règles. Cependant, il est plus courant qu'il y ait une relation un à un entre un DLQ et un gestionnaire DLQ.

IBM i *Vérification du traitement de tous les messages DLQ sur IBM i*

Le gestionnaire de files d'attente des messages non livrés conserve un enregistrement de tous les messages de la file d'attente des messages non livrés IBM i qui ont été vus mais qui n'ont pas été supprimés.

Si vous utilisez le gestionnaire DLQ comme filtre pour extraire un petit sous-ensemble des messages du DLQ, le gestionnaire DLQ conserve toujours un enregistrement de ces messages sur le DLQ qu'il n'a pas traité. De plus, le gestionnaire DLQ ne peut pas garantir que les nouveaux messages arrivant sur le DLQ

seront vus, même si le DLQ est défini comme premier entré, premier sorti (FIFO). Si la file d'attente n'est pas vide, le DLQ est périodiquement réanalysé pour vérifier tous les messages.

Pour ces raisons, assurez-vous que le DLQ contient le moins de messages possible. Si des messages qui ne peuvent pas être supprimés ou réacheminés vers d'autres files d'attente (pour quelque raison que ce soit) sont autorisés à s'accumuler dans la file d'attente, la charge de travail du gestionnaire DLQ augmente et la file d'attente DLQ elle-même risque de se remplir.

Vous pouvez prendre des mesures spécifiques pour permettre au gestionnaire DLQ de vider le DLQ. Par exemple, essayez de ne pas utiliser ACTION (IGNORE), qui laisse des messages sur le DLQ. (N'oubliez pas que ACTION (IGNORE) est utilisé pour les messages qui ne sont pas explicitement adressés par d'autres règles de la table.) A la place, pour les messages que vous ignorez, utilisez une action qui déplace les messages vers une autre file d'attente. Exemple :

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

De même, faites de la règle finale de la table un fourre-tout pour traiter les messages qui n'ont pas été traités par les règles précédentes de la table. Par exemple, la règle finale de la table peut ressembler à ceci:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Ainsi, les messages qui tombent dans la règle finale de la table sont réacheminés vers la file d'attente REALLY.DEAD.QUEUE, où ils peuvent être traités manuellement. Si vous n'avez pas de règle de ce type, les messages risquent de rester indéfiniment dans la file d'attente des messages non livrés.

IBM i

Exemple de table de règles de gestionnaire DLQ sous IBM i

Exemple de code pour une table de règles de gestionnaire de files d'attente de rebut sous IBM i. Cet exemple de table de règles contient une entrée de données de contrôle unique et plusieurs règles.

```
*****
*   An example rules table for the STRMQMDLQ command   *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* STRMQMDLQ, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to STRMQMDLQ,
* use the DLQ defined for the local queue manager.
*
inputqm(' ') inputq(' ')

* Rules
* -----
* We include rules with ACTION (RETRY) first to try to
* deliver the message to the intended destination.

* If a message is placed on the DLQ because its destination
* queue is full, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_Q_FULL) ACTION(RETRY) RETRY(5)

* If a message is placed on the DLQ because of a put inhibited
* condition, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)

* The AAAA corporation is always sending messages with incorrect
* addresses. When we find a request from the AAAA corporation,
* we return it to the DLQ (DEADQ) of the reply-to queue manager
* (&REPLYQM).
* The AAAA DLQ handler attempts to redirect the message.
```

```
MSGTYPE(MQMT_REQUEST) REPLYQM(AAAA.*) +
ACTION(FWD) FWDQ(DEADQ) FWDQM(&REPLYQM)
```

* The BBBB corporation never does things by half measures. If
* the queue manager BBBB.1 is unavailable, try to
* send the message to BBBB.2

```
DESTQM(bbbb.1) +
action(fwd) fwdq(&DESTQ) fwdqm(bbbb.2) header(no)
```

* The CCCC corporation considers itself very security
* conscious, and believes that none of its messages
* will ever end up on one of our DLQs.
* Whenever we see a message from a CCCC queue manager on our
* DLQ, we send it to a special destination in the CCCC organization
* where the problem is investigated.

```
REPLYQM(CCCC.*) +
ACTION(FWD) FWDQ(ALARM) FWDQM(CCCC.SYSTEM)
```

* Messages that are not persistent run the risk of being
* lost when a queue manager terminates. If an application
* is sending nonpersistent messages, it must be able
* to cope with the message being lost, so we can afford to
* discard the message.

```
PERSIST(MQPER_NOT_PERSISTENT) ACTION(DISCARD)
```

* For performance and efficiency reasons, we like to keep
* the number of messages on the DLQ small.
* If we receive a message that has not been processed by
* an earlier rule in the table, we assume that it
* requires manual intervention to resolve the problem.
* Some problems are best solved at the node where the
* problem was detected, and others are best solved where
* the message originated. We do not have the message origin,
* but we can use the REPLYQM to identify a node that has
* some interest in this message.
* Attempt to put the message onto a manual intervention
* queue at the appropriate node. If this fails,
* put the message on the manual intervention queue at
* this node.

```
REPLYQM('?*') +
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION) FWDQM(&REPLYQM)
```

```
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION)
```

Appel du gestionnaire DLQ

Appelez le gestionnaire de files d'attente de rebut à l'aide de la commande `runmqdlq`. Vous pouvez nommer le DLQ que vous souhaitez traiter et le gestionnaire de files d'attente que vous souhaitez utiliser de deux manières.

Les deux méthodes sont les suivantes:

- En tant que paramètres de `runmqdlq` à partir de l'invite de commande. Exemple :

```
runmqdlq ABC1.DEAD.LETTER.QUEUE ABC1.QUEUE.MANAGER <qrule.rul
```

- Dans la table de règles. Exemple :

```
INPUTQ(ABC1.DEAD.LETTER.QUEUE) INPUTQM(ABC1.QUEUE.MANAGER)
```

Les exemples s'appliquent à la file d'attente des messages non livrés appelée `ABC1.DEAD.LETTER.QUEUE`, détenue par le gestionnaire de files d'attente `ABC1.QUEUE.MANAGER`.

Si vous ne spécifiez pas le DLQ ou le gestionnaire de files d'attente comme indiqué, le gestionnaire de files d'attente par défaut pour l'installation est utilisé avec le DLQ appartenant à ce gestionnaire de files d'attente.

La commande `runmqdlq` extrait son entrée de `stdin`; vous associez la table de règles à `runmqdlq` en redirigeant `stdin` à partir de la table de règles.

Pour exécuter le gestionnaire DLQ, vous devez être autorisé à accéder à la fois à la file d'attente DLQ elle-même et aux files d'attente de messages vers lesquelles les messages de la file d'attente DLQ sont réacheminés. Pour que le gestionnaire DLQ puisse placer des messages dans des files d'attente avec les droits de l'ID utilisateur dans le contexte de message, vous devez également être autorisé à assumer l'identité des autres utilisateurs.

Pour plus d'informations sur la commande `runmqdlq`, voir [runmqdlq](#).

Information associée

[Files d'attente de rebut](#)

[Traitement des incidents liés aux messages non distribués](#)

Exemple de gestionnaire DLQ `amqsdlq`

En plus du gestionnaire de files d'attente de rebut appelé à l'aide de la commande `runmqdlq`, IBM MQ fournit la source d'un exemple de gestionnaire DLQ `amqsdlq` avec une fonction similaire à celle fournie par `runmqdlq`.

Vous pouvez personnaliser `amqsdlq` pour fournir un gestionnaire DLQ qui répond à vos besoins. Par exemple, vous pouvez décider que vous souhaitez un gestionnaire DLQ pouvant traiter des messages sans en-têtes de rebut. (Le gestionnaire DLQ par défaut et l'exemple, `amqsdlq`, ne traitent que les messages sur le DLQ qui commencent par un en-tête de rebut, MQDLH. Les messages qui ne commencent pas par un MQDLH sont identifiés comme étant erronés et restent indéfiniment sur le DLQ.)

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

Dans IBM MQ for Windows, la source de `amqsdlq` est fournie dans le répertoire:

```
MQ_INSTALLATION_PATH\tools\c\samples\dlq
```

et la version compilée est fournie dans le répertoire:

```
MQ_INSTALLATION_PATH\tools\c\samples\bin
```

Sur les systèmes IBM MQ for UNIX et Linux, la source d'`amqsdlq` est fournie dans le répertoire:

```
MQ_INSTALLATION_PATH/samp/dlq
```

et la version compilée est fournie dans le répertoire:

```
MQ_INSTALLATION_PATH/samp/bin
```

Vous pouvez également compiler `amqsdlq` en mode client. Pour plus d'informations, voir [Ecriture d'applications de procédure client](#), [Génération d'applications pour IBM MQ MQI clientset](#) et [Exécution d'applications dans l'environnement IBM MQ MQI client](#).

Table des règles du gestionnaire DLQ

La table des règles du gestionnaire de files d'attente de rebut définit la façon dont le gestionnaire de files d'attente de rebut traite les messages qui arrivent sur la file d'attente de rebut.

Il existe deux types d'entrée dans une table de règles:

- La première entrée de la table, qui est facultative, contient des *données de contrôle*.
- Toutes les autres entrées de la table sont des *règles* que le gestionnaire DLQ doit suivre. Chaque règle se compose d'un *modèle* (un ensemble de caractéristiques de message) auquel un message est comparé et d'une *action* à effectuer lorsqu'un message sur la file d'attente des messages non livrés correspond au modèle spécifié. Une table de règles doit contenir au moins une règle.

Chaque entrée de la table de règles comprend un ou plusieurs mots clés.

Information associée

[Files d'attente de rebut](#)

[Traitement des incidents liés aux messages non distribués](#)

Données de contrôle DLQ

Vous pouvez inclure des mots clés dans une entrée de données de contrôle dans une table de règles de gestionnaire de files d'attente de rebut.

Remarque :

- La ligne verticale (|) sépare les alternatives, dont une seule peut être spécifiée.
- Tous les mots clés sont facultatifs.

INPUTQ (QueueName|')

Nom de la file d'attente des données à traiter:

1. Toute valeur INPUTQ que vous fournissez en tant que paramètre à la commande `runmqdlq` remplace toute valeur INPUTQ dans la table de règles.
2. Si vous ne spécifiez pas de valeur INPUTQ comme paramètre de la commande `runmqdlq`, mais que vous **indiquez** une valeur dans la table de règles, la valeur INPUTQ de la table de règles est utilisée.
3. Si aucun DLQ n'est spécifié ou que vous spécifiez INPUTQ (") dans la table de règles, le nom du DLQ appartenant au gestionnaire de files d'attente avec le nom fourni en tant que paramètre à la commande `runmqdlq` est utilisé.
4. Si vous ne spécifiez pas de valeur INPUTQ comme paramètre de la commande `runmqdlq` ou comme valeur dans la table de règles, le DLQ appartenant au gestionnaire de files d'attente nommé sur le mot clé INPUTQM dans la table de règles est utilisé.

INPUTQM (QueueManagerNom|')

Nom du gestionnaire de files d'attente propriétaire de la file d'attente des messages non livrés indiquée dans le mot clé INPUTQ:

1. Toute valeur INPUTQM que vous fournissez en tant que paramètre à la commande `runmqdlq` remplace toute valeur INPUTQM dans la table de règles.
2. Si vous ne spécifiez pas de valeur INPUTQM comme paramètre de la commande `runmqdlq`, la valeur INPUTQM de la table de règles est utilisée.
3. Si aucun gestionnaire de files d'attente n'est spécifié ou si vous spécifiez INPUTQM (") dans la table de règles, le gestionnaire de files d'attente par défaut pour l'installation est utilisé.

RETRYINT (Intervalle|60)

Intervalle, en secondes, auquel le gestionnaire DLQ doit retraiter les messages sur le DLQ qui n'ont pas pu être traités lors de la première tentative et pour lesquels des tentatives répétées ont été demandées. Par défaut, l'intervalle entre les nouvelles tentatives est de 60 secondes.

WAIT (YES|NO|nnn)

Indique si le gestionnaire DLQ doit attendre que d'autres messages arrivent sur le DLQ lorsqu'il détecte qu'il ne peut pas traiter d'autres messages.

YES

Le gestionnaire DLQ attend indéfiniment.

NO

Le gestionnaire DLQ s'arrête lorsqu'il détecte que le DLQ est vide ou qu'il ne contient aucun message qu'il peut traiter.

nnn

Le gestionnaire DLQ attend *nnn* secondes pour que le nouveau travail arrive avant de s'arrêter, après avoir détecté que la file d'attente est vide ou qu'elle ne contient aucun message qu'il peut traiter.

Indiquez WAIT (YES) pour les DLQ occupés et WAIT (NO) ou WAIT (*nnn*) pour les DLQ dont le niveau d'activité est faible. Si le gestionnaire DLQ est autorisé à s'arrêter, appelez-le à nouveau à l'aide du déclenchement. Pour plus d'informations sur le déclenchement, voir [Démarriage des applications IBM MQ à l'aide de déclencheurs](#).

Une alternative à l'inclusion de données de contrôle dans la table de règles consiste à fournir les noms du DLQ et de son gestionnaire de files d'attente en tant que paramètres d'entrée à la commande `runmqdlq`. Si vous spécifiez une valeur à la fois dans la table de règles et en tant qu'entrée de la commande `runmqdlq`, la valeur spécifiée dans la commande `runmqdlq` est prioritaire.

Si vous incluez une entrée de données de contrôle dans la table de règles, il doit s'agir de la **première** entrée de la table.

Règles DLQ (modèles et actions)

Description des mots clés de correspondance de modèle (ceux auxquels correspondent les messages de la file d'attente des messages non livrés) et des mots clés d'action (ceux qui déterminent comment le gestionnaire DLQ doit traiter un message correspondant). Un exemple de règle est également fourni.

Mots clés de correspondance de modèle

Les mots clés de correspondance de modèle, que vous utilisez pour spécifier les valeurs auxquelles les messages du DLQ sont mis en correspondance, sont les suivants. (Tous les mots clés de correspondance de modèle sont facultatifs):

APPLIDAT (*DonnéesApplIdentity* | *)

Valeur *ApplIdentityData* spécifiée dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

APPLNAME (*PutApplNom* | *)

Nom de l'application qui a émis l'appel MQPUT ou MQPUT1, comme indiqué dans la zone *PutApplNom* du descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

APPLTYPE (*PutApplType* | *)

Valeur *PutApplType*, spécifiée dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

DESTQ (*QueueName* | *)

Nom de la file d'attente de messages à laquelle le message est destiné.

DESTQM (*QueueManagerNom* | *)

Nom du gestionnaire de files d'attente de la file d'attente de messages à laquelle le message est destiné.

FEEDBACK (*Commentaires en retour* | *)

Lorsque la valeur de *MsgType* est MQFB_REPORT, *Feedback* décrit la nature du rapport.

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQFB_COA pour identifier les messages sur le DLQ qui ont besoin d'une confirmation de leur arrivée dans leurs files d'attente de destination.

FORMAT (*Format* | *)

Nom utilisé par l'expéditeur du message pour décrire le format des données du message.

MSGTYPE (*MsgType* | *)

Type du message sur la file d'attente des messages non livrés.

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQMT_REQUEST pour identifier les messages sur le DLQ qui ont besoin de réponses.

PERSISTANT (*Persistence* | *)

Valeur de persistance du message. (La persistance d'un message détermine s'il survit aux redémarrages du gestionnaire de files d'attente.)

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQPER_PERSISTENT pour identifier les messages sur le DLQ qui sont persistants.

REASON (*ReasonCode* | *)

Code raison décrivant la raison pour laquelle le message a été inséré dans le DLQ.

Vous pouvez utiliser des noms symboliques. Par exemple, vous pouvez utiliser le nom symbolique MQRC_Q_FULL pour identifier les messages placés dans la file d'attente des messages non livrés car leurs files d'attente de destination étaient saturées.

REPLYQ (*QueueName* | *)

Nom de la file d'attente de réponses indiquée dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

REPLYQM (*QueueManagerNom* | *)

Nom du gestionnaire de files d'attente de la file d'attente de réponse, tel qu'indiqué dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

USERID (*UserIdentifier* | *)

ID utilisateur de l'utilisateur à l'origine du message sur la file d'attente des messages non livrés, comme indiqué dans le descripteur de message, MQMD, du message sur la file d'attente des messages non livrés.

Les mots clés d'action

Les mots clés d'action utilisés pour décrire la façon dont un message correspondant doit être traité sont les suivants:

ACTION (DISCARD | IGNORE | RETRY | FWD)

Action à effectuer pour tout message de la file d'attente des messages non livrés qui correspond au modèle défini dans cette règle.

SUPPRIMER

Supprimez le message du DLQ.

IGNORE

Laissez le message sur le DLQ.

REESSAYER

Si la première tentative d'insertion du message dans sa file d'attente de destination échoue, réessayez. Le mot clé RETRY définit le nombre de tentatives effectuées pour implémenter une action. Le mot clé RETRYINT des données de contrôle contrôle l'intervalle entre les tentatives.

Transfert

Réachemine le message vers la file d'attente indiquée dans le mot clé FWDQ.

Vous devez spécifier le mot clé ACTION.

FWDQ (*QueueName* | & DESTQ | & REPLYQ)

Nom de la file d'attente de messages vers laquelle le message doit être réacheminé lorsque la commande ACTION (FWD) est demandée.

QueueName

Nom d'une file d'attente de messages. FWDQ (") n'est pas valide.

& DESTQ

Prenez le nom de la file d'attente dans la zone *DestQName* de la structure MQDLH.

& REPLYQ

Prenez le nom de la file d'attente dans la zone *ReplyToQ* du descripteur de message, MQMD.

Pour éviter les messages d'erreur lorsqu'une règle spécifiant FWDQ (& REPLYQ) correspond à un message dont la zone *ReplyToQ* est vide, spécifiez REPLYQ (? *) dans le modèle de message.

FWDQM (*QueueManagerNom* | & DESTQM | & REPLYQM | ' ')

Gestionnaire de files d'attente de la file d'attente vers laquelle transférer un message.

QueueManagerName

Nom du gestionnaire de files d'attente de la file d'attente vers laquelle transférer un message lorsque la commande ACTION (FWD) est demandée.

& DESTQM

Utilisez le nom du gestionnaire de files d'attente dans la zone *DestQMgrNom* de la structure MQDLH.

& REPLYQM

Utilisez le nom du gestionnaire de files d'attente dans la zone *ReplyToQMgr* du descripteur de message, MQMD.

..

FWDQM ('), qui est la valeur par défaut, identifie le gestionnaire de files d'attente local.

HEADER (OUI|NON)

Indique si le MQDLH doit rester sur un message pour lequel ACTION (FWD) est demandée. Par défaut, le MQDLH reste sur le message. Le mot clé HEADER n'est pas valide pour les actions autres que FWD.

PUTAUT (DEF| CTX)

Droits avec lesquels les messages doivent être insérés par le gestionnaire DLQ:

DEF

Placez les messages avec les droits du gestionnaire DLQ lui-même.

CTX

Placez les messages avec les droits de l'ID utilisateur dans le contexte de message. Si vous indiquez PUTAUT (CTX), vous devez être autorisé à assumer l'identité d'autres utilisateurs.

RETRY (*Nombre de nouvelles tentatives*|1)

Nombre de fois, compris entre 1 et 999,999,999, où une action doit être exécutée (à l'intervalle indiqué dans le mot clé RETRYINT des données de contrôle). Le nombre de tentatives effectuées par le gestionnaire de file d'attente des messages non livrés pour implémenter une règle particulière est spécifique à l'instance en cours du gestionnaire de file d'attente des messages non livrés ; ce nombre n'est pas conservé lors des redémarrages. Si le gestionnaire DLQ est redémarré, le nombre de tentatives d'application d'une règle est réinitialisé à zéro.

Exemple de règle

Voici un exemple de règle d'une table de règles de gestionnaire DLQ:

```
PERSIST(MQPER_PERSISTENT) REASON (MQRC_PUT_INHIBITED) +
ACTION (RETRY) RETRY (3)
```

Cette règle demande au gestionnaire de file d'attente des messages non livrés d'effectuer trois tentatives de distribution à sa file d'attente de destination de tout message persistant inséré dans la file d'attente des messages non livrés car MQPUT et MQPUT1 ont été inhibés.

Tous les mots clés que vous pouvez utiliser sur une règle sont décrits dans le reste de cette section. Notez ce qui suit :

- La valeur par défaut d'un mot clé, le cas échéant, est soulignée. Pour la plupart des mots clés, la valeur par défaut est * (astérisque), qui correspond à n'importe quelle valeur.
- La ligne verticale (|) sépare les alternatives, dont une seule peut être spécifiée.
- Tous les mots clés sauf ACTION sont facultatifs.

Conventions de la table de règles DLQ

La syntaxe, la structure et le contenu de la table des règles du gestionnaire de files d'attente de rebut doivent respecter ces conventions.

La table de règles doit respecter les conventions suivantes:

- Une table de règles doit contenir au moins une règle.
- Les mots clés peuvent apparaître dans n'importe quel ordre.
- Un mot clé ne peut être inclus qu'une seule fois dans une règle.
- Les mots clés ne sont pas sensibles à la casse.

- Un mot clé et sa valeur de paramètre doivent être séparés des autres mots clés par au moins un espace ou une virgule.
- Il peut y avoir n'importe quel nombre de blancs au début ou à la fin d'une règle, et entre les mots clés, la ponctuation et les valeurs.
- Chaque règle doit commencer sur une nouvelle ligne.
- Sur les systèmes Windows, la dernière règle de la table doit se terminer par un retour chariot / saut de ligne. Pour ce faire, vous devez appuyer sur la touche Entrée à la fin de la règle, de sorte que la dernière ligne de la table soit une ligne vide.
- Pour des raisons de portabilité, la longueur significative d'une ligne ne doit pas dépasser 72 caractères.
- Utilisez le signe plus (+) comme dernier caractère non blanc sur une ligne pour indiquer que la règle se poursuit à partir du premier caractère non blanc sur la ligne suivante. Utilisez le signe moins (-) comme dernier caractère non blanc sur une ligne pour indiquer que la règle se poursuit à partir du début de la ligne suivante. Les caractères de continuation peuvent apparaître dans les mots clés et les paramètres.

Exemple :

```
APPLNAME(' ABC+
D')
```

résultats dans 'ABCD', et

```
APPLNAME(' ABC-
D')
```

génère 'ABC D'.

- Les lignes de commentaire, qui commencent par un astérisque (*), peuvent apparaître n'importe où dans la table de règles.
- Les lignes vides sont ignorées.
- Chaque entrée de la table de règles du gestionnaire DLQ comprend un ou plusieurs mots clés et leurs paramètres associés. Les paramètres doivent respecter les règles de syntaxe suivantes:
 - Chaque valeur de paramètre doit comporter au moins un caractère significatif. Les guillemets simples délimiteurs dans les valeurs placées entre guillemets ne sont pas considérés comme significatifs. Par exemple, les paramètres suivants sont valides:

| | |
|----------------|----------------------------|
| FORMAT(' ABC') | 3 caractères significatifs |
| FORMAT(ABC) | 3 caractères significatifs |
| FORMAT(' A') | 1 caractère significatif |
| FORMAT(A) | 1 caractère significatif |
| FORMAT(' ') | 1 caractère significatif |

Ces paramètres ne sont pas valides car ils ne contiennent pas de caractères significatifs:

```
FORMAT(' ' )
FORMAT( )
FORMAT( )
FORMAT
```

- Les caractères génériques sont pris en charge. Vous pouvez utiliser le point d'interrogation (?) au lieu de n'importe quel caractère unique, à l'exception d'un blanc de fin ; vous pouvez utiliser l'astérisque (*) au lieu de zéro ou plusieurs caractères adjacents. L'astérisque (*) et le point d'interrogation (?) sont **toujours** interprétés comme des caractères génériques dans les valeurs de paramètre.

- Les caractères génériques ne peuvent pas être inclus dans les paramètres des mots clés suivants: ACTION, HEADER, RETRY, FWDQ, FWDQM et PUTAUT.
- Les blancs de fin dans les valeurs de paramètre et dans les zones correspondantes du message sur le DLQ ne sont pas importants lors de l'exécution de correspondances de caractères génériques. Toutefois, les blancs de début et les blancs imbriqués dans les chaînes qui sont placés entre apostrophes sont importants pour les correspondances de caractères génériques.
- Les paramètres numériques ne peuvent pas inclure le caractère générique point d'interrogation (?). Vous pouvez utiliser l'astérisque (*) à la place d'un paramètre numérique entier, mais pas dans le cadre d'un paramètre numérique. Par exemple, les paramètres numériques suivants sont valides:

| | |
|---------------|--|
| MSGTYPE (2) | Seuls les messages de réponse sont éligibles |
| MSGTYPE (*) | Tout type de message est éligible |
| MSGTYPE ('*') | Tout type de message est éligible |

Toutefois, MSGTYPE ('2*') n'est pas valide car il inclut un astérisque (*) dans un paramètre numérique.

- Les paramètres numériques doivent être compris entre 0 et 999 999 999. Si la valeur de paramètre est comprise dans cette plage, elle est acceptée, même si elle n'est pas actuellement valide dans la zone à laquelle le mot clé est associé. Vous pouvez utiliser des noms symboliques pour les paramètres numériques.
- Si une valeur de chaîne est plus courte que la zone de la MQDLH ou de la MQMD à laquelle le mot clé se rapporte, la valeur est remplie avec des blancs à la longueur de la zone. Si la valeur, à l'exclusion des astérisques, est plus longue que la zone, une erreur est diagnostiquée. Par exemple, voici toutes les valeurs de chaîne valides pour une zone de 8 caractères:

| | |
|---------------------|--|
| 'ABCDEFGH' | 8 caractères |
| 'A*C*E*G*I' | 5 caractères à l'exclusion des astérisques |
| '*A*C*E*G*I*K*M*O*' | 8 caractères à l'exclusion des astérisques |

- Placez entre apostrophes les chaînes contenant des blancs, des minuscules ou des caractères spéciaux autres que le point (.), la barre oblique (?), le trait de soulignement (_) et le signe de pourcentage (%). Les caractères minuscules qui ne sont pas placés entre apostrophes sont convertis en majuscules. Si la chaîne inclut un guillemet, utilisez deux apostrophes pour indiquer à la fois le début et la fin de l'apostrophe. Lorsque la longueur de la chaîne est calculée, chaque occurrence de guillemets est comptée comme un caractère unique.

Mode de traitement de la table de règles DLQ

Le gestionnaire de files d'attente de rebut recherche dans la table de règles une règle dans laquelle le modèle correspond à un message sur le DLQ.

La recherche commence par la première règle de la table et se poursuit séquentiellement dans la table. Lorsque le gestionnaire DLQ trouve une règle avec un modèle correspondant, il effectue l'action à partir de cette règle. Le gestionnaire DLQ incrémente le nombre de nouvelles tentatives pour une règle de 1 chaque fois qu'il applique cette règle. Si la première tentative échoue, le gestionnaire de la file d'attente des messages non livrés effectue une nouvelle tentative jusqu'à ce que le nombre de tentatives corresponde au nombre indiqué dans le mot clé RETRY. Si toutes les tentatives échouent, le gestionnaire DLQ recherche la règle de correspondance suivante dans la table.

Ce processus est répété pour les règles de correspondance suivantes jusqu'à ce qu'une action aboutisse. Lorsque chaque règle correspondante a été tentée le nombre de fois indiqué dans son mot clé RETRY et que toutes les tentatives ont échoué, ACTION (IGNORE) est utilisé. ACTION (IGNORE) est également pris en compte si aucune règle de correspondance n'est trouvée.

Remarque :

1. Les modèles de règle de correspondance sont recherchés uniquement pour les messages de la file d'attente des messages non livrés qui commencent par un MQDLH. Les messages qui ne commencent pas par un MQDLH sont signalés périodiquement comme étant erronés et restent indéfiniment sur le DLQ.
2. Tous les mots clés de modèle peuvent être autorisés par défaut, de sorte qu'une règle ne peut être constituée que d'une action. Notez toutefois que les règles d'action uniquement sont appliquées à tous les messages de la file d'attente qui ont des MQDLH et qui n'ont pas déjà été traités conformément aux autres règles de la table.
3. La table de règles est validée lorsque le gestionnaire DLQ démarre et les erreurs sont signalées à ce moment-là. Vous pouvez apporter des modifications à la table de règles à tout moment, mais ces modifications n'entrent en vigueur que lorsque le gestionnaire DLQ redémarre.
4. Le gestionnaire DLQ ne modifie pas le contenu des messages, le MQDLH ou le descripteur de message. Le gestionnaire DLQ insère toujours des messages dans d'autres files d'attente avec l'option de message MQPMO_PASS_ALL_CONTEXT.
5. Les erreurs de syntaxe consécutives dans la table de règles peuvent ne pas être reconnues car la table de règles est conçue pour éliminer la génération d'erreurs répétitives lors de la validation.
6. Le gestionnaire DLQ ouvre le DLQ avec l'option MQOO_INPUT_AS_Q_DEF.
7. Plusieurs instances du gestionnaire DLQ peuvent s'exécuter simultanément sur la même file d'attente, à l'aide de la même table de règles. Cependant, il est plus courant qu'il y ait une relation un à un entre un DLQ et un gestionnaire DLQ.

Information associée

Files d'attente de rebut

Traitement des incidents liés aux messages non distribués

Vérification du traitement de tous les messages DLQ

Le gestionnaire de file d'attente de rebut conserve un enregistrement de tous les messages de la file d'attente des messages non livrés qui ont été vus mais qui n'ont pas été supprimés.

Si vous utilisez le gestionnaire DLQ comme filtre pour extraire un petit sous-ensemble des messages du DLQ, le gestionnaire DLQ doit toujours conserver un enregistrement de ces messages sur le DLQ qu'il n'a pas traité. De plus, le gestionnaire DLQ ne peut pas garantir que les nouveaux messages arrivant sur le DLQ sont vus, même si le DLQ est défini comme premier entré, premier sorti (FIFO). Si la file d'attente n'est pas vide, le DLQ est périodiquement réanalysé pour vérifier tous les messages.

Pour ces raisons, essayez de vous assurer que le DLQ contient le moins de messages possible ; si des messages qui ne peuvent pas être supprimés ou réacheminés vers d'autres files d'attente (pour quelque raison que ce soit) sont autorisés à s'accumuler dans la file d'attente, la charge de travail du gestionnaire DLQ augmente et le DLQ lui-même peut se remplir.

Vous pouvez prendre des mesures spécifiques pour permettre au gestionnaire DLQ de vider le DLQ. Par exemple, essayez de ne pas utiliser ACTION (IGNORE), qui laisse des messages sur le DLQ. (N'oubliez pas que ACTION (IGNORE) est utilisé pour les messages qui ne sont pas explicitement adressés par d'autres règles de la table.) A la place, pour les messages que vous ignorez, utilisez une action qui déplace les messages vers une autre file d'attente. Exemple :

```
ACTION (FWD) FWDQ (IGNORED.DEAD.QUEUE) HEADER (YES)
```

De même, faites de la règle finale de la table un fourre-tout pour traiter les messages qui n'ont pas été traités par les règles précédentes de la table. Par exemple, la règle finale de la table peut ressembler à ceci:

```
ACTION (FWD) FWDQ (REALLY.DEAD.QUEUE) HEADER (YES)
```

Les messages qui tombent dans la règle finale de la table sont ainsi réacheminés vers la file d'attente REALLY . DEAD . QUEUE, où ils peuvent être traités manuellement. Si vous ne disposez pas d'une telle règle, les messages risquent de rester indéfiniment dans la file d'attente des messages non livrés.

Exemple de table de règles de gestionnaire DLQ

Exemple de table de règles de file d'attente de rebut pour la commande `runmqdlq`, contenant une seule entrée de données de contrôle et plusieurs règles.

```
*****
*   An example rules table for the runmqdlq command   *
*****
* Control data entry
* -----
* If no queue manager name is supplied as an explicit parameter to
* runmqdlq, use the default queue manager for the machine.
* If no queue name is supplied as an explicit parameter to runmqdlq,
* use the DLQ defined for the local queue manager.
*
inputqm(' ') inputq(' ')

* Rules
* -----
* We include rules with ACTION (RETRY) first to try to
* deliver the message to the intended destination.
* If a message is placed on the DLQ because its destination
* queue is full, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_Q_FULL) ACTION(RETRY) RETRY(5)

* If a message is placed on the DLQ because of a put inhibited
* condition, attempt to forward the message to its
* destination queue. Make 5 attempts at approximately
* 60-second intervals (the default value for RETRYINT).

REASON(MQRC_PUT_INHIBITED) ACTION(RETRY) RETRY(5)

* The AAAA corporation are always sending messages with incorrect
* addresses. When we find a request from the AAAA corporation,
* we return it to the DLQ (DEADQ) of the reply-to queue manager
* (&REPLYQM).
* The AAAA DLQ handler attempts to redirect the message.

MSGTYPE(MQMT_REQUEST) REPLYQM(AAAA.*) +
ACTION(FWD) FWDQ(DEADQ) FWDQM(&REPLYQM)

* The BBBB corporation never do things by half measures. If
* the queue manager BBBB.1 is unavailable, try to
* send the message to BBBB.2

DESTQM(bbbb.1) +
action(fwd) fwdq(&DESTQ) fwdqm(bbbb.2) header(no)

* The CCCC corporation considers itself very security
* conscious, and believes that none of its messages
* will ever end up on one of our DLQs.
* Whenever we see a message from a CCCC queue manager on our
* DLQ, we send it to a special destination in the CCCC organization
* where the problem is investigated.

REPLYQM(CCCC.*) +
ACTION(FWD) FWDQ(ALARM) FWDQM(CCCC.SYSTEM)
```

```
* Messages that are not persistent run the risk of being
* lost when a queue manager terminates. If an application
* is sending nonpersistent messages, it should be able
* to cope with the message being lost, so we can afford to
* discard the message. PERSIST(MQPER_NOT_PERSISTENT) ACTION(DISCARD)
* For performance and efficiency reasons, we like to keep
* the number of messages on the DLQ small.
* If we receive a message that has not been processed by
* an earlier rule in the table, we assume that it
* requires manual intervention to resolve the problem.
* Some problems are best solved at the node where the
* problem was detected, and others are best solved where
* the message originated. We don't have the message origin,
* but we can use the REPLYQM to identify a node that has
* some interest in this message.
* Attempt to put the message onto a manual intervention
* queue at the appropriate node. If this fails,
```

```
* put the message on the manual intervention queue at
* this node.

REPLYQM('?*') +
ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION) FWDQM(&REPLYQM)

ACTION(FWD) FWDQ(DEADQ.MANUAL.INTERVENTION)
```

Information associée

Files d'attente de rebut

[Traitement des incidents liés aux messages non distribués](#)

[runmqdlq \(exécution du gestionnaire de files d'attente de rebut\)](#)

Utilisation des files d'attente modèles

Un gestionnaire de files d'attente crée une *file d'attente dynamique* s'il reçoit un appel MQI d'une application spécifiant un nom de file d'attente qui a été défini comme file d'attente modèle. Le nom de la nouvelle file d'attente dynamique est généré par le gestionnaire de files d'attente lors de la création de la file d'attente. Une *file d'attente modèle* est un modèle qui spécifie les attributs des files d'attente dynamiques créées à partir de celle-ci. Les files d'attente modèles offrent aux applications une méthode pratique pour créer des files d'attente selon les besoins.

Définition d'une file d'attente modèle

La commande **DEFINE QMODEL** permet de définir une file d'attente modèle avec un ensemble d'attributs de la même manière que vous définissez une file d'attente locale. Les files d'attente modèles et les files d'attente locales ont le même ensemble d'attributs, sauf que dans les files d'attente modèles, vous pouvez indiquer si les files d'attente dynamiques créées sont temporaires ou permanentes. (Les files d'attente permanentes sont gérées lors des redémarrages du gestionnaire de files d'attente, alors que les files d'attente temporaires ne le sont pas.) Exemple :

```
DEFINE QMODEL (GREEN.MODEL.QUEUE) +
DESCR('Queue for messages from application X') +
PUT (DISABLED) +
GET (ENABLED) +
NOTRIGGER +
MSGDLVSQ (FIFO) +
MAXDEPTH (1000) +
MAXMSGL (2000) +
USAGE (NORMAL) +
DEFTYPE (PERMDYN)
```

Cette commande crée une définition de file d'attente modèle. A partir de l'attribut **DEFTYPE**, vous pouvez voir que les files d'attente réelles créées à partir de ce modèle sont des files d'attente dynamiques permanentes. Tous les attributs non spécifiés sont automatiquement copiés à partir de `SYSDTEM.DEFAULT.MODEL.QUEUE`.

Vous pouvez utiliser les attributs **LIKE** et **REPLACE** lorsque vous définissez des files d'attente modèles, de la même manière que vous les utilisez avec des files d'attente locales.

Utilisation d'autres commandes avec des files d'attente modèles

Vous pouvez utiliser les commandes MQSC appropriées pour afficher ou modifier les attributs d'une file d'attente modèle ou pour supprimer l'objet de file d'attente modèle. Exemple :

Utilisez la commande **DISPLAY QUEUE** pour afficher les attributs de la file d'attente modèle:

```
DISPLAY QUEUE (GREEN.MODEL.QUEUE)
```

Utilisez la commande **ALTER QMODEL** pour modifier le modèle afin d'activer les insertions dans n'importe quelle file d'attente dynamique créée à partir de ce modèle:

```
ALTER QMODEL (BLUE.MODEL.QUEUE) PUT(ENABLED)
```

Utilisez la commande **DELETE QMODEL** pour supprimer cette file d'attente modèle:

```
DELETE QMODEL (RED.MODEL.QUEUE)
```

Information associée

[ALTER QMODEL](#)

[DEFINIR QMODEL](#)

[SUPPRIMER QMODEL](#)

[DISPLAY QUEUE](#)

Utilisation des rubriques d'administration

Utilisez les commandes MQSC pour gérer les rubriques d'administration.

Pour plus d'informations sur ces commandes, voir [Commandes MQSC](#).

Concepts associés

«Définition d'une rubrique d'administration», à la page 179

Utilisez la commande MQSC **DEFINE TOPIC** pour créer une rubrique d'administration. Lorsque vous définissez une rubrique d'administration, vous pouvez éventuellement définir chaque attribut de rubrique.

«Affichage des attributs d'objet de rubrique d'administration», à la page 180

La commande MQSC **DISPLAY TOPIC** permet d'afficher un objet de rubrique d'administration.

«Modification des attributs de rubrique d'administration», à la page 181

Vous pouvez modifier les attributs de rubrique de deux manières, à l'aide de la commande **ALTER TOPIC** ou de la commande **DEFINE TOPIC** avec l'attribut **REPLACE**.

«Copie d'une définition de rubrique d'administration», à la page 181

Vous pouvez copier une définition de rubrique à l'aide de l'attribut **LIKE** de la commande **DEFINE**.

«Suppression d'une définition de rubrique d'administration», à la page 182

Vous pouvez utiliser la commande MQSC **DELETE TOPIC** pour supprimer une rubrique d'administration.

Information associée

[Objets de rubrique d'administration](#)

Définition d'une rubrique d'administration

Utilisez la commande MQSC **DEFINE TOPIC** pour créer une rubrique d'administration. Lorsque vous définissez une rubrique d'administration, vous pouvez éventuellement définir chaque attribut de rubrique.

Tout attribut de la rubrique qui n'est pas explicitement défini est hérité de la rubrique d'administration par défaut, **SYSTEM.DEFAULT.TOPIC**, qui a été créé lors de l'installation du système.

Par exemple, la commande **DEFINE TOPIC** qui suit, définit une rubrique appelée **ORANGE.TOPIC** avec les caractéristiques suivantes:

- Se résout en la chaîne de rubrique **ORANGE**. Pour plus d'informations sur la façon dont les chaînes de rubrique peuvent être utilisées, voir [Combinaison de chaînes de rubrique](#).
- Tout attribut défini sur **ASPARENT** utilise l'attribut tel que défini par la rubrique parent de cette rubrique. Cette action est répétée dans l'arborescence de rubriques jusqu'à la rubrique racine, **SYSTEM.BASE.TOPIC** a été trouvé. Pour plus d'informations, voir [Arborescences de rubriques](#).

```
DEFINE TOPIC (ORANGE.TOPIC) +  
TOPICSTR (ORANGE) +
```

```
DEFPRTY(ASPARENT) +
NPMSGDLV(ASPARENT)
```

Remarque :

- A l'exception de la valeur de la chaîne de rubrique, toutes les valeurs d'attribut affichées sont les valeurs par défaut. Ils ne sont présentés ici qu'à titre d'illustration. Vous pouvez les omettre si vous êtes sûr que les valeurs par défaut sont celles que vous souhaitez ou qu'elles n'ont pas été modifiées. Voir aussi «Affichage des attributs d'objet de rubrique d'administration», à la page 180.
- Si vous disposez déjà d'une rubrique d'administration sur le même gestionnaire de files d'attente avec le nom ORANGE.TOPIC, cette commande échoue. Utilisez l'attribut REPLACE si vous souhaitez remplacer la définition existante d'une rubrique, mais voir aussi «Modification des attributs de rubrique d'administration», à la page 181.

Information associée

[DEFINE TOPIC](#)

Affichage des attributs d'objet de rubrique d'administration

La commande MQSC **DISPLAY TOPIC** permet d'afficher un objet de rubrique d'administration.

Pour afficher toutes les rubriques, utilisez:

```
DISPLAY TOPIC(ORANGE.TOPIC)
```

Vous pouvez afficher les attributs de manière sélective en les spécifiant individuellement à l'aide de la commande **DISPLAY TOPIC**. Exemple :

```
DISPLAY TOPIC(ORANGE.TOPIC) +
TOPICSTR +
DEFPRTY +
NPMSGDLV
```

Cette commande affiche les trois attributs spécifiés comme suit:

```
AMQ8633: Display topic details.
TOPIC(ORANGE.TOPIC)
TOPICSTR(ORANGE)
NPMSGDLV(ASPARENT)
TYPE(LOCAL)
DEFPRTY(ASPARENT)
```

Pour afficher les valeurs ASPARENT de la rubrique telles qu'elles sont utilisées lors de l'exécution, utilisez la commande **DISPLAY TPSTATUS**. Par exemple, utilisez :

```
DISPLAY TPSTATUS(ORANGE) DEFPRTY NPMSGDLV
```

La commande affiche les détails suivants:

```
AMQ8754: Display topic status details.
TOPICSTR(ORANGE)
NPMSGDLV(ALLAVAIL)
DEFPRTY(0)
```

Lorsque vous définissez une rubrique d'administration, elle prend tous les attributs que vous ne spécifiez pas explicitement à partir de la rubrique d'administration par défaut, appelée SYSTEM.DEFAULT.TOPIC. Pour voir quels sont ces attributs par défaut, utilisez la commande suivante:

```
DISPLAY TOPIC (SYSTEM.DEFAULT.TOPIC)
```

Information associée

[DISPLAY TOPIC](#)

[DISPLAY TPSTATUS](#)

Modification des attributs de rubrique d'administration

Vous pouvez modifier les attributs de rubrique de deux manières, à l'aide de la commande **ALTER TOPIC** ou de la commande **DEFINE TOPIC** avec l'attribut **REPLACE**.

Par exemple, si vous souhaitez modifier la priorité par défaut des messages distribués à une rubrique appelée ORANGE.TOPIC, à la valeur 5, utilisez l'une des commandes suivantes.

- A l'aide de la commande **ALTER** :

```
ALTER TOPIC(ORANGE.TOPIC) DEFPRTY(5)
```

Cette commande modifie à 5 un attribut unique, celui de la priorité par défaut des messages distribués à cette rubrique ; tous les autres attributs restent identiques.

- A l'aide de la commande **DEFINE** :

```
DEFINE TOPIC(ORANGE.TOPIC) DEFPRTY(5) REPLACE
```

Cette commande modifie la priorité par défaut des messages distribués à cette rubrique. Tous les autres attributs reçoivent leurs valeurs par défaut.

Si vous modifiez la priorité des messages envoyés à cette rubrique, les messages existants ne sont pas affectés. Tout nouveau message, cependant, utilise la priorité spécifiée si elle n'est pas fournie par l'application de publication.

Information associée

[ALTER TOPIC](#)

[DISPLAY TOPIC](#)

Copie d'une définition de rubrique d'administration

Vous pouvez copier une définition de rubrique à l'aide de l'attribut **LIKE** de la commande **DEFINE**.

Exemple :

```
DEFINE TOPIC (MAGENTA.TOPIC) +  
LIKE (ORANGE.TOPIC)
```

Cette commande crée une rubrique, MAGENTA.TOPIC, avec les mêmes attributs que la rubrique d'origine, ORANGE.TOPIC, plutôt que ceux de la rubrique d'administration par défaut du système. Entrez le nom de la rubrique à copier exactement tel qu'il a été entré lors de la création de la rubrique. Si le nom contient des caractères minuscules, placez-le entre apostrophes.

Vous pouvez également utiliser cette forme de la commande **DEFINE** pour copier une définition de rubrique, mais apporter des modifications aux attributs de l'original. Exemple :

```
DEFINE TOPIC(BLUE.TOPIC) +  
TOPICSTR(BLUE) +  
LIKE(ORANGE.TOPIC)
```

Vous pouvez également copier les attributs de la rubrique BLUE.TOPIC à la rubrique GREEN.TOPIC et indiquez que lorsque les publications ne peuvent pas être distribuées à leur file d'attente de souscription correcte, elles ne sont pas placées dans la file d'attente de rebut. Exemple :

```
DEFINE TOPIC(GREEN.TOPIC) +  
TOPICSTR(GREEN) +  
LIKE(BLUE.TOPIC) +  
USEDLQ(NO)
```

Information associée

[DEFINE TOPIC](#)

Suppression d'une définition de rubrique d'administration

Vous pouvez utiliser la commande MQSC **DELETE TOPIC** pour supprimer une rubrique d'administration.

Exemple :

```
DELETE TOPIC(ORANGE.TOPIC)
```

Les applications ne pourront plus ouvrir la rubrique pour la publication ou créer de nouveaux abonnements à l'aide du nom d'objet ORANGE.TOPIC. Les applications de publication pour lesquelles la rubrique est ouverte peuvent continuer à publier la chaîne de rubrique résolue. Tous les abonnements déjà effectués à cette rubrique continuent de recevoir des publications après la suppression de la rubrique.

Les applications qui ne font pas référence à cet objet de rubrique mais qui utilisent la chaîne de rubrique résolue représentée par cet objet de rubrique, 'ORANGE' dans cet exemple, continuent de fonctionner. Dans ce cas, ils héritent des propriétés d'un objet de rubrique plus haut dans l'arborescence de rubriques. Pour plus d'informations, voir [Arborescences de rubriques](#).

Information associée

[Supprimer la rubrique](#)

Utilisation des abonnements

Utilisez les commandes MQSC pour gérer les abonnements.

Les abonnements peuvent être de l'un des trois types définis dans l'attribut **SUBTYPE** :

ADMIN

Défini administrativement par un utilisateur.

Proxy

Abonnement créé en interne pour le routage des publications entre les gestionnaires de files d'attente.

API

Créé à l'aide d'un programme, par exemple, à l'aide de l'appel MQI MQSUB.

Pour plus d'informations sur ces commandes, voir [Commandes MQSC](#).

Concepts associés

[«Définition d'un abonnement d'administration», à la page 182](#)

Utilisez la commande MQSC **DEFINE SUB** pour créer un abonnement d'administration. Vous pouvez également utiliser la valeur par défaut définie dans la définition de l'abonnement local par défaut. Vous pouvez également modifier les caractéristiques de l'abonnement à partir de celles de l'abonnement local par défaut, SYSTEM.DEFAULT.SUB créée lors de l'installation du système.

[«Affichage des attributs des abonnements», à la page 183](#)

Vous pouvez utiliser la commande **DISPLAY SUB** pour afficher les attributs configurés de tout abonnement connu du gestionnaire de files d'attente.

[«Modification des attributs d'abonnement local», à la page 184](#)

Vous pouvez modifier les attributs d'abonnement de deux manières, à l'aide de la commande **ALTER SUB** ou de la commande **DEFINE SUB** avec l'attribut **REPLACE**.

[«Copie d'une définition d'abonnement local», à la page 185](#)

Vous pouvez copier une définition d'abonnement à l'aide de l'attribut **LIKE** de la commande **DEFINE**.

[«Suppression d'un abonnement local», à la page 185](#)

Vous pouvez utiliser la commande MQSC **DELETE SUB** pour supprimer un abonnement local.

Définition d'un abonnement d'administration

Utilisez la commande MQSC **DEFINE SUB** pour créer un abonnement d'administration. Vous pouvez également utiliser la valeur par défaut définie dans la définition de l'abonnement local par défaut. Vous

pouvez également modifier les caractéristiques de l'abonnement à partir de celles de l'abonnement local par défaut, SYSTEM.DEFAULT.SUB créée lors de l'installation du système.

Par exemple, la commande **DEFINE SUB** qui suit définit un abonnement appelé ORANGE avec les caractéristiques suivantes:

- Abonnement durable, ce qui signifie qu'il est conservé après le redémarrage du gestionnaire de files d'attente, avec un délai d'expiration illimité.
- Réception des publications effectuées sur la chaîne de rubrique ORANGE, avec les priorités de message définies par les applications de publication.
- Les publications fournies pour cet abonnement sont envoyées à la file d'attente locale SUBQ. Cette file d'attente doit être définie avant la définition de l'abonnement.

```
DEFINE SUB (ORANGE) +
TOPICSTR (ORANGE) +
DESTCLAS (PROVIDED) +
DEST (SUBQ) +
EXPIRY (UNLIMITED) +
PUBPRTY (ASPUB)
```

Remarque :

- L'abonnement et le nom de la chaîne de rubrique ne doivent pas nécessairement correspondre.
- A l'exception des valeurs de la destination et de la chaîne de rubrique, toutes les valeurs d'attribut affichées sont les valeurs par défaut. Ils ne sont présentés ici qu'à titre d'illustration. Vous pouvez les omettre si vous êtes sûr que les valeurs par défaut sont celles que vous souhaitez ou qu'elles n'ont pas été modifiées. Voir aussi «Affichage des attributs des abonnements», à la page 183.
- Si vous disposez déjà d'un abonnement local sur le même gestionnaire de files d'attente avec le nom ORANGE, cette commande échoue. Utilisez l'attribut **REPLACE** si vous souhaitez remplacer la définition existante d'une file d'attente, mais voir aussi «Modification des attributs d'abonnement local», à la page 184.
- Si la file d'attente SUBQ n'existe pas, cette commande échoue.

Information associée

DEFINE SUB

Affichage des attributs des abonnements

Vous pouvez utiliser la commande **DISPLAY SUB** pour afficher les attributs configurés de tout abonnement connu du gestionnaire de files d'attente.

Par exemple, utilisez :

```
DISPLAY SUB(ORANGE)
```

Vous pouvez afficher les attributs de manière sélective en les spécifiant individuellement. Exemple :

```
DISPLAY SUB(ORANGE) +
SUBID +
TOPICSTR +
DURABLE
```

Cette commande affiche les trois attributs spécifiés comme suit:

```
AMQ8096: IBM MQ subscription inquired.
SUBID(414D5120414120202020202020202020EE921E4E20002A03)
SUB(ORANGE)
DURABLE(YES)
TOPICSTR(ORANGE)
```

TOPICSTR est la chaîne de rubrique résolue sur laquelle cet abonné fonctionne. Lorsqu'un abonnement est défini pour utiliser un objet de rubrique, la chaîne de rubrique de cet objet est utilisée comme préfixe de la chaîne de rubrique fournie lors de la création de l'abonnement. SUBID est un identificateur unique affecté par le gestionnaire de files d'attente lors de la création d'un abonnement. Il s'agit d'un attribut utile à afficher car certains noms d'abonnement peuvent être longs ou dans des jeux de caractères différents pour lesquels il peut devenir peu pratique.

Une autre méthode d'affichage des abonnements consiste à utiliser le SUBID:

```
DISPLAY SUB +
SUBID(414D51204141412020202020202020EE921E4E20002A03) +
TOPICSTR +
DURABLE
```

Cette commande génère la même sortie que précédemment:

```
AMQ8096: IBM MQ subscription inquired.
SUBID(414D512041414120202020202020EE921E4E20002A03)
SUB(ORANGE)
DURABLE(YES)
TOPICSTR(ORANGE)
```

Les abonnements de proxy sur un gestionnaire de files d'attente ne sont pas affichés par défaut. Pour les afficher, spécifiez un **SUBTYPE** de type PROXY ou ALL.

Vous pouvez utiliser la commande [DISPLAY SBSTATUS](#) pour afficher les attributs d'exécution. Par exemple, utilisez la commande suivante:

```
DISPLAY SBSTATUS(ORANGE) NUMMSGS
```

La sortie suivante s'affiche :

```
AMQ8099: IBM MQ subscription status inquired.
SUB(ORANGE)
SUBID(414D512041414120202020202020EE921E4E20002A03)
NUMMSGS(0)
```

Lorsque vous définissez un abonnement d'administration, il prend tous les attributs que vous ne spécifiez pas explicitement à partir de l'abonnement par défaut, appelé SYSTEM.DEFAULT.SUB. Pour voir quels sont ces attributs par défaut, utilisez la commande suivante:

```
DISPLAY SUB (SYSTEM.DEFAULT.SUB)
```

Information associée

[DISPLAY SUB](#)

Modification des attributs d'abonnement local

Vous pouvez modifier les attributs d'abonnement de deux manières, à l'aide de la commande **ALTER SUB** ou de la commande **DEFINE SUB** avec l'attribut **REPLACE**.

Si, par exemple, vous souhaitez modifier la priorité des messages distribués à un abonnement appelé ORANGE sur 5, utilisez l'une des commandes suivantes:

- A l'aide de la commande **ALTER** :

```
ALTER SUB(ORANGE) PUBPRTY(5)
```

Cette commande modifie un attribut unique, celui de la priorité des messages distribués à cet abonnement à 5 ; tous les autres attributs restent les mêmes.

- A l'aide de la commande **DEFINE** :

```
DEFINE SUB(ORANGE) PUBPRTY(5) REPLACE
```

Cette commande modifie non seulement la priorité des messages distribués à cet abonnement, mais également tous les autres attributs auxquels sont affectées leurs valeurs par défaut.

Si vous modifiez la priorité des messages envoyés à cet abonnement, les messages existants ne sont pas affectés. Toutefois, tous les nouveaux messages ont la priorité spécifiée.

Information associée

[ALTER SUB](#)

[DEFINE SUB](#)

Copie d'une définition d'abonnement local

Vous pouvez copier une définition d'abonnement à l'aide de l'attribut **LIKE** de la commande **DEFINE** .

Exemple :

```
DEFINE SUB(BLUE) +  
LIKE(ORANGE)
```

Vous pouvez également copier les attributs de la sous-commande REAL dans la sous-commande THIRD.SUB et indiquer que l' correlID des publications distribuées est THIRD, plutôt que l' correlID des diffuseurs de publications. Exemple :

```
DEFINE SUB(THIRD.SUB) +  
LIKE(BLUE) +  
DESTCORL(ORANGE)
```

Information associée

[DEFINE SUB](#)

Suppression d'un abonnement local

Vous pouvez utiliser la commande MQSC **DELETE SUB** pour supprimer un abonnement local.

```
DELETE SUB(ORANGE)
```

Vous pouvez également supprimer un abonnement à l'aide du SUBID:

```
DELETE SUB SUBID(414D51204141412020202020202020EE921E4E20002A03)
```

Information associée

[SUPPRIMER DES SOUS](#)

Vérification des messages sur un abonnement

Lorsqu'un abonnement est défini, il est associé à une file d'attente. Les messages publiés correspondant à cet abonnement sont placés dans cette file d'attente.

Pourquoi et quand exécuter cette tâche

Notez que les commandes **runmqsc** suivantes affichent uniquement les abonnements qui ont reçu des messages.

Pour rechercher les messages actuellement mis en file d'attente pour un abonnement, procédez comme suit:

Procédure

1. Pour rechercher les messages mis en file d'attente pour un type d'abonnement DISPLAY SBSTATUS(*sub_name*) NUMMSGs, voir «Affichage des attributs des abonnements», à la page 183.
2. Si la valeur **NUMMSGs** est supérieure à zéro, identifiez la file d'attente associée à l'abonnement en entrant DISPLAY SUB(*sub_name*) DEST.
3. En utilisant le nom de la file d'attente renvoyée, vous pouvez afficher les messages en suivant la technique décrite dans «Exploration des files d'attente», à la page 153.

Information associée

STATUT SBSTATUS D'AFFICHAGE

Utilisation des services

Les objets de service sont un moyen par lequel des processus supplémentaires peuvent être gérés dans le cadre d'un gestionnaire de files d'attente. Avec les services, vous pouvez définir des programmes qui sont démarrés et arrêtés lorsque le gestionnaire de files d'attente démarre et se termine. Les services IBM MQ sont toujours démarrés sous l'ID de l'utilisateur qui a démarré le gestionnaire de files d'attente.

Pour définir une nouvelle définition de service IBM MQ, utilisez la commande MQSC DEFINE SERVICE.

Les objets de service peuvent être de l'un des types suivants:

serveur

Un serveur est un objet de service dont le paramètre **SERVTYPE** est spécifié comme SERVER. Un objet de service serveur est la définition d'un programme exécuté lors du démarrage d'un gestionnaire de files d'attente spécifié. Les objets de service serveur définissent des programmes qui s'exécutent généralement pendant une longue période. Par exemple, un objet de service de serveur peut être utilisé pour exécuter un processus de moniteur de déclenchement, tel que **runmqtrm**.

Une seule instance d'un objet de service de serveur peut s'exécuter simultanément. Le statut des objets de service du serveur en cours d'exécution peut être surveillé à l'aide de la commande MQSC, **DISPLAY SVSTATUS**.

Commande

Une commande est un objet de service dont le paramètre **SERVTYPE** est spécifié sous la forme COMMAND. Les objets de service de commande sont similaires aux objets de service de serveur, mais plusieurs instances d'un objet de service de commande peuvent s'exécuter simultanément et leur statut ne peut pas être surveillé à l'aide de la commande MQSC **DISPLAY SVSTATUS**.

Si la commande MQSC, **STOP SERVICE**, est exécutée, aucune vérification n'est effectuée pour déterminer si le programme démarré par la commande MQSC, **START SERVICE**, est toujours actif avant l'arrêt du programme.

Information associée

STATUT DE L'AFFICHAGE

Démarrage du service

Arrêt du service

Définition d'un objet de service

Vous définissez un objet de service à l'aide de la commande MQSC **DEFINE SERVICE**.

Les attributs que vous devez définir sont les suivants:

SERVTYPE

Définit le type de l'objet de service. Les valeurs admises sont les suivantes :

SERVEUR

Un objet de service de serveur.

Une seule instance d'un objet de service de serveur peut être exécutée à la fois. Le statut des objets de service du serveur peut être surveillé à l'aide de la commande MQSC, **DISPLAY SVSTATUS**.

Commande

Objet de service de commande.

Plusieurs instances d'un objet de service de commande peuvent être exécutées simultanément. L'état des objets de service de commande ne peut pas être surveillé.

STARTCMD

Programme exécuté pour démarrer le service. Vous devez indiquer un chemin d'accès complet au programme.

STARTARG

Arguments transmis au programme de démarrage.

STDERR

Indique le chemin d'accès à un fichier vers lequel l'erreur standard (stderr) du programme de service doit être redirigée.

STDOUT

Indique le chemin d'accès à un fichier vers lequel la sortie standard (stdout) du programme de service doit être redirigée.

STOPCMD

Programme exécuté pour arrêter le service. Vous devez indiquer un chemin d'accès complet au programme.

STOPARG

Arguments transmis au programme d'arrêt.

CONTROL

Indique comment le service doit être démarré et arrêté:

MANUAL

Le service ne doit pas être démarré automatiquement ou arrêté automatiquement. Il est contrôlé par l'utilisation des commandes **START SERVICE** et **STOP SERVICE**. Il s'agit de la valeur par défaut.

QMGR

Le service en cours de définition doit être démarré et arrêté en même temps que le gestionnaire de files d'attente est démarré et arrêté.

STARTONLY (UNIQUEMENT)

Le service doit être démarré en même temps que le gestionnaire de files d'attente, mais il n'est pas demandé de s'arrêter lorsque le gestionnaire de files d'attente est arrêté.

Concepts associés

«Gestion des services», à la page 187

A l'aide du paramètre **CONTROL**, une instance d'un objet de service peut être démarrée et arrêtée automatiquement par le gestionnaire de files d'attente, ou démarrée et arrêtée à l'aide des commandes MQSC **START SERVICE** et **STOP SERVICE**.

Information associée

[Définir un service](#)

[STATUT DE L'AFFICHAGE](#)

[Démarrage du service](#)

[Arrêt du service](#)

Gestion des services

A l'aide du paramètre **CONTROL**, une instance d'un objet de service peut être démarrée et arrêtée automatiquement par le gestionnaire de files d'attente, ou démarrée et arrêtée à l'aide des commandes MQSC **START SERVICE** et **STOP SERVICE**.

Lorsqu'une instance d'un objet de service est démarrée, un message contenant le nom de l'objet de service et l'ID de processus du processus démarré est consigné dans le journal des erreurs du gestionnaire de files d'attente. Voici un exemple d'entrée de journal pour le démarrage d'un objet de service de serveur:

```
02/15/2005 11:54:24 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5028: The Server 'S1' has started. ProcessId(13031).

EXPLANATION:
The Server process has started.
ACTION:
None.
```

Voici un exemple d'entrée de journal pour le démarrage d'un objet de service de commande:

```
02/15/2005 11:53:55 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5030: The Command 'C1' has started. ProcessId(13030).

EXPLANATION:
The Command has started.
ACTION:
None.
```

Lorsqu'un service de serveur d'instance s'arrête, un message est consigné dans les journaux d'erreurs du gestionnaire de files d'attente contenant le nom du service et l'ID de processus du processus en cours de fin. Voici un exemple d'entrée de journal pour l'arrêt d'un objet de service de serveur:

```
02/15/2005 11:54:54 AM - Process(10363.1) User(mqm) Program(amqzmgr0)
Host(HOST_1) Installation(Installation1)
VRMF(7.1.0.0) QMgr(A.B.C)
AMQ5029: The Server 'S1' has ended. ProcessId(13031).

EXPLANATION:
The Server process has ended.
ACTION:
None.
```

Référence associée

«Autres variables d'environnement», à la page 188

L'environnement dans lequel le processus d'un service est démarré est hérité de celui du gestionnaire de files d'attente. Il est possible de définir des variables d'environnement supplémentaires à définir dans l'environnement du processus de service en ajoutant les variables à définir à l'un des fichiers de substitution d'environnement `service.env`.

Information associée

[Arrêt du service](#)

[Démarrage du service](#)

Autres variables d'environnement

L'environnement dans lequel le processus d'un service est démarré est hérité de celui du gestionnaire de files d'attente. Il est possible de définir des variables d'environnement supplémentaires à définir dans l'environnement du processus de service en ajoutant les variables à définir à l'un des fichiers de substitution d'environnement `service.env`.

Fichiers auxquels vous pouvez ajouter des variables d'environnement

Il existe deux fichiers possibles auxquels vous pouvez ajouter des variables d'environnement:

Le fichier `service.env` de la portée de la machine

Ce fichier se trouve dans:

- **Linux** **UNIX** /var/mqm sur les systèmes UNIX and Linux .
- **Windows** Répertoire de données sélectionné lors de l'installation sur les systèmes Windows .

Fichier `service.env` de portée du gestionnaire de files d'attente

Ce fichier se trouve dans le répertoire de données du gestionnaire de files d'attente. Par exemple, l'emplacement du fichier de substitution d'environnement pour un gestionnaire de files d'attente nommé QMNAME est:

- – **Linux** **UNIX** Sur les systèmes UNIX and Linux , /var/mqm/qmgrs/QMNAME/service.env
- **Windows** Sur les systèmes Windows ,
C:\ProgramData\IBM\MQ\qmgrs\QMNAME\service.env

Les deux fichiers sont traités, s'ils sont disponibles, avec les définitions du fichier de portée du gestionnaire de files d'attente prioritaires sur celles du fichier de portée de la machine.

Variables d'environnement pouvant être spécifiées dans `service.env`.

Toute variable d'environnement peut être spécifiée dans `service.env`. Par exemple, si le service IBM MQ exécute un certain nombre de commandes, il peut être utile de définir la variable utilisateur `PATH` dans le fichier `service.env`. Les valeurs que vous définissez pour la variable ne peuvent pas être des variables d'environnement ; par exemple, `CLASSPATH= %CLASSPATH%` est incorrect. De même, sur Linux `PATH= $PATH : /opt/mqm/bin`, des résultats inattendus sont générés.

`CLASSPATH` doit être en majuscules et l'instruction de chemin d'accès aux classes ne peut contenir que des littéraux. Certains services (télémétrie, par exemple) définissent leur propre chemin d'accès aux classes. La variable `CLASSPATH` définie dans le fichier `service.env` est ajoutée à ce dernier.

Le format des variables définies dans le fichier `service.env` est une liste de paires de variables de nom et de valeur. Chaque variable doit être définie sur une nouvelle ligne, et chaque variable est prise telle qu'elle est explicitement définie, y compris les espaces.

Exemple de `service.env`

```

#*****#
##*                                     *##
##* <N_OCO_COPYRIGHT>                 *##
##* Licensed Materials - Property of IBM *##
##*                                     *##
##* 63H9336                             *##
##* (C) Copyright IBM Corporation 2005, 2023. *##
##*                                     *##
##* <NOC_COPYRIGHT>                   *##
##*                                     *##
#*****#
##* Module Name: service.env           *##
##* Type       : IBM MQ service environment file *##
##* Function   : Define additional environment variables to be set *##
##*           : for SERVICE programs. *##
##* Usage     : <VARIABLE>=<VALUE> *##
##*           : *##
#*****#
MYLOC=/opt/myloc/bin
MYTMP=/tmp
TRACEDIR=/tmp/trace
MYINITQ=ACCOUNTS.INITIATION.QUEUE

```

Référence associée

«Insertions remplaçables dans les définitions de service», à la page 190

Dans la définition d'un objet de service, il est possible de remplacer des jetons. Les jetons qui sont remplacés sont automatiquement remplacés par leur texte développé lors de l'exécution du programme

de service. Les jetons de substitution peuvent être extraits de la liste suivante de jetons communs ou de toute variable définie dans le fichier, service.env.

Insertions remplaçables dans les définitions de service

Dans la définition d'un objet de service, il est possible de remplacer des jetons. Les jetons qui sont remplacés sont automatiquement remplacés par leur texte développé lors de l'exécution du programme de service. Les jetons de substitution peuvent être extraits de la liste suivante de jetons communs ou de toute variable définie dans le fichier, service.env.

Les jetons communs suivants peuvent être utilisés pour remplacer des jetons dans la définition d'un objet de service:

CHEMIN_INSTALL_MQ

Emplacement dans lequel IBM MQ est installé.

MQ_DATA_PATH

Emplacement du répertoire de données IBM MQ :

- **Linux** **UNIX** Sur les systèmes UNIX and Linux , l'emplacement du répertoire de données IBM MQ est /var/mqm/
- **Windows** Sur les systèmes Windows , l'emplacement du répertoire de données IBM MQ est le répertoire de données sélectionné lors de l'installation de IBM MQ

QMNAME

Nom du gestionnaire de files d'attente en cours.

NOM_SERVICE_MQ

Nom du service.

PID SERVEUR_MQ

Ce jeton ne peut être utilisé que par les arguments **STOPARG** et **STOPCMD** .

Pour les objets de service de serveur, ce jeton est remplacé par l'ID de processus du processus démarré par les arguments **STARTCMD** et **STARTARG** . Sinon, ce jeton est remplacé par 0.

CHEMIN_DATA_MQ_Q_MGR_

Emplacement du répertoire de données du gestionnaire de files d'attente.

MQ_Q_MGR_DATA_NAME

Nom transformé du gestionnaire de files d'attente. Pour plus d'informations sur la transformation de nom, voir [Description des noms de fichier IBM MQ](#).

Pour utiliser des insertions remplaçables, insérez le jeton dans les caractères + dans les chaînes **STARTCMD**, **STARTARG**, **STOPCMD**, **STOPARG**, **STDOUT** ou **STDERR** . Pour des exemples, voir [«Exemples d'utilisation d'objets de service»](#), à la page 190.

Exemples d'utilisation d'objets de service

Les services de cette section sont écrits avec des caractères de séparation de chemin de style UNIX , sauf indication contraire.

Utilisation d'un objet de service de serveur

Cet exemple montre comment définir, utiliser et modifier un objet de service de serveur pour démarrer un moniteur de déclenchement.

1. Un objet de service serveur est défini à l'aide de la commande **DEFINE SERVICE** MQSC:

```
DEFINE SERVICE(S1) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+bin/runmqtrm') +
STARTARG('-m +QMNAME+ -q ACCOUNTS.INITIATION.QUEUE') +
STOPCMD('+MQ_INSTALL_PATH+bin/amqsstop') +
STOPARG('-m +QMNAME+ -p +MQ_SERVER_PID+')
```

où :

+MQ_INSTALL_PATH+ est un jeton représentant le répertoire d'installation.

+QMNAME+ est un jeton représentant le nom du gestionnaire de files d'attente.

ACCOUNTS.INITIATION.QUEUE est la file d'attente d'initialisation.

amqsstop est un exemple de programme fourni avec IBM MQ qui demande au gestionnaire de files d'attente d'interrompre toutes les connexions pour l'ID de processus. amqsstop génère des commandes PCF. Par conséquent, le serveur de commandes doit être en cours d'exécution.

+MQ_SERVER_PID+ est un jeton représentant l'ID de processus transmis au programme d'arrêt.

Pour obtenir la liste des jetons communs, voir [«Insertions remplaçables dans les définitions de service»](#), à la page 190 .

2. Une instance de l'objet de service du serveur s'exécute lors du prochain démarrage du gestionnaire de files d'attente. Toutefois, nous démarrons une instance de l'objet de service serveur immédiatement à l'aide de la commande **START SERVICE MQSC**:

```
START SERVICE(S1)
```

3. Le statut du processus de service du serveur est affiché à l'aide de la commande **DISPLAY SVSTATUS MQSC**:

```
DISPLAY SVSTATUS(S1)
```

4. Cet exemple montre maintenant comment modifier l'objet de service du serveur et faire en sorte que les mises à jour soient prises en compte en redémarrant manuellement le processus de service du serveur. L'objet de service du serveur est modifié de sorte que la file d'attente d'initialisation soit spécifiée sous la forme JUPITER.INITIATION.QUEUE. La commande **ALTER SERVICE MQSC** est utilisée:

```
ALTER SERVICE(S1) +  
STARTARG('-m +QMNAME+ -q JUPITER.INITIATION.QUEUE')
```

Remarque : Un service en cours d'exécution ne récupère aucune mise à jour de sa définition de service tant qu'il n'est pas redémarré.

5. Le processus de service du serveur est redémarré afin que la modification soit prise en compte, à l'aide des commandes MQSC **STOP SERVICE** et **START SERVICE** :

```
STOP SERVICE(S1)
```

suivie de :

```
START SERVICE(S1)
```

Le processus de service du serveur est redémarré et prend en compte les modifications apportées dans «4», à la page 191.

Remarque : La commande MQSC, **STOP SERVICE**, ne peut être utilisée que si un argument **STOPCMD** est spécifié dans la définition de service.

Information associée

[MODIFIER LE SERVICE](#)

[Définir un service](#)

[STATUT DE L'AFFICHAGE](#)

[Démarrage du service](#)

[Arrêt du service](#)

Utilisation d'un objet de service de commande

Cet exemple montre comment définir un objet de service de commande pour démarrer un programme qui écrit des entrées dans le journal système du système d'exploitation lorsqu'un gestionnaire de files d'attente est démarré ou arrêté.

1. L'objet de service de commande est défini à l'aide de la commande **DEFINE SERVICE** MQSC:

```
DEFINE SERVICE(S2) +  
CONTROL(QMGR) +  
SERVTYPE(COMMAND) +  
STARTCMD('/usr/bin/logger') +  
STARTARG('Queue manager +QMNAME+ starting') +  
STOPCMD('/usr/bin/logger') +  
STOPARG('Queue manager +QMNAME+ stopping')
```

où :

logger est la commande fournie par le système UNIX and Linux pour écrire dans le journal système.

+QMNAME+ est un jeton représentant le nom du gestionnaire de files d'attente.

Information associée

[Définir un service](#)

Utilisation d'un objet de service de commande lorsqu'un gestionnaire de files d'attente se termine uniquement

Cet exemple montre comment définir un objet de service de commande pour démarrer un programme qui écrit des entrées dans le journal système du système d'exploitation lorsqu'un gestionnaire de files d'attente est arrêté uniquement.

1. L'objet de service de commande est défini à l'aide de la commande **DEFINE SERVICE** MQSC:

```
DEFINE SERVICE(S3) +  
CONTROL(QMGR) +  
SERVTYPE(COMMAND) +  
STOPCMD('/usr/bin/logger') +  
STOPARG('Queue manager +QMNAME+ stopping')
```

où :

logger est un exemple de programme fourni avec IBM MQ qui peut écrire des entrées dans le journal système du système d'exploitation.

+QMNAME+ est un jeton représentant le nom du gestionnaire de files d'attente.

Information associée

[Définir un service](#)

Plus sur la transmission d'arguments

Cet exemple montre comment définir un objet de service de serveur pour démarrer un programme appelé runserv lorsqu'un gestionnaire de files d'attente est démarré.

Cet exemple est écrit avec des caractères de séparation de chemin de style Windows .

L'un des arguments à transmettre au programme de démarrage est une chaîne contenant un espace. Cet argument doit être transmis sous la forme d'une chaîne unique. Pour ce faire, des guillemets sont utilisés comme indiqué dans la commande suivante pour définir l'objet de service de commande:

1. L'objet de service serveur est défini à l'aide de la commande **DEFINE SERVICE** MQSC:

```
DEFINE SERVICE(S1) SERVTYPE(SERVER) CONTROL(QMGR) +  
STARTCMD('C:\Program Files\Tools\runserv.exe') +  
STARTARG('-m +QMNAME+ -d "C:\Program Files\Tools\"') +  
STDOUT('C:\Program Files\Tools\+MQ_SERVICE_NAME+.out')
```



```

DEFINE SERVICE(S4) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('C:\Program Files\Tools\runserv.exe') +
STARTARG('-m +QMNAME+ -d "C:\Program Files\Tools\'') +
STDOUT('C:\Program Files\Tools\+MQ_SERVICE_NAME+.out')

```

où :

+QMNAME+ est un jeton représentant le nom du gestionnaire de files d'attente.

"C:\Program Files\Tools\" est une chaîne contenant un espace, qui sera transmise sous la forme d'une chaîne unique.

Information associée

[Définir un service](#)

Démarrage automatique d'un service

Cet exemple montre comment définir un objet de service de serveur qui peut être utilisé pour démarrer automatiquement le moniteur de déclenchement lorsque le gestionnaire de files d'attente démarre.

1. L'objet de service serveur est défini à l'aide de la commande **DEFINE SERVICE** MQSC:

```

DEFINE SERVICE(TRIG_MON_START) +
CONTROL(QMGR) +
SERVTYPE(SERVER) +
STARTCMD('runmqtrm') +
STARTARG('-m +QMNAME+ -q +IQNAME+')

```

où :

+QMNAME+ est un jeton représentant le nom du gestionnaire de files d'attente.

+IQNAME+ est une variable d'environnement définie par l'utilisateur dans l'un des fichiers service.env représentant le nom de la file d'attente d'initialisation.

Information associée

[Définir un service](#)

Gestion des objets pour le déclenchement

IBM MQ vous permet de démarrer automatiquement une application lorsque certaines conditions d'une file d'attente sont remplies. Par exemple, vous pouvez démarrer une application lorsque le nombre de messages d'une file d'attente atteint un nombre spécifié. Cette fonction est appelée *déclenchement*. Vous devez définir les objets qui prennent en charge le déclenchement.

Déclenchement décrit en détail dans [Démarrage des applications IBM MQ à l'aide de déclencheurs](#).

Définition d'une file d'attente d'application pour le déclenchement

Une file d'attente d'application est une file d'attente locale utilisée par les applications pour la messagerie, via l'interface MQI. Le déclenchement requiert la définition d'un certain nombre d'attributs de file d'attente dans la file d'attente d'application.

Le déclenchement lui-même est activé par l'attribut **Trigger** (TRIGGER dans les commandes MQSC). Dans cet exemple, un événement déclencheur doit être généré lorsqu'il y a 100 messages de priorité 5 ou supérieure dans la file d'attente locale MOTOR.INSURANCE.QUEUE, comme suit:

```

DEFINE QLOCAL (MOTOR.INSURANCE.QUEUE) +
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +
MAXMSGL (2000) +
DEFPSIST (YES) +
INITQ (MOTOR.INS.INIT.QUEUE) +
TRIGGER +
TRIGTYPE (DEPTH) +
TRIGDPH (100)+
TRIGMPRI (5)

```

où :

QLOCAL (MOTOR.INSURANCE.QUEUE)

Nom de la file d'attente d'application en cours de définition.

PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)

Nom de la définition de processus qui définit l'application à démarrer par un programme de moniteur de déclenchement.

MAXMSGL (2000)

Longueur maximale des messages dans la file d'attente.

DEFPSIST (YES)

Indique que les messages de cette file d'attente sont persistants par défaut.

INITQ (MOTOR.INS.INIT.QUEUE)

Nom de la file d'attente d'initialisation dans laquelle le gestionnaire de files d'attente doit insérer le message de déclenchement.

TRIGGER

Valeur de l'attribut de déclencheur.

TRIGTYPE (DEPTH)

Indique qu'un événement déclencheur est généré lorsque le nombre de messages de priorité requise (TRIGMPRI) atteint le nombre indiqué dans TRIGDPTH.

TRIGDPTH (100)

Nombre de messages requis pour générer un événement déclencheur.

TRIGMPRI (5)

Priorité des messages qui doivent être comptés par le gestionnaire de files d'attente pour déterminer s'il convient de générer un événement déclencheur. Seuls les messages de priorité 5 ou supérieure sont comptés.

Définition d'une file d'attente d'initialisation

Lorsqu'un événement déclencheur se produit, le gestionnaire de files d'attente place un message de déclenchement dans la file d'attente d'initialisation spécifiée dans la définition de file d'attente d'application. Les files d'attente d'initialisation n'ont pas de paramètres spéciaux, mais vous pouvez utiliser la définition suivante de la file d'attente locale MOTOR.INS.INIT.QUEUE pour obtenir des conseils:

```
DEFINE QLOCAL(MOTOR.INS.INIT.QUEUE) +
GET (ENABLED) +
NOSHARE +
NOTRIGGER +
MAXMSGL (2000) +
MAXDEPTH (1000)
```

Définition d'un processus

Utilisez la commande DEFINE PROCESS pour créer une définition de processus. Une définition de processus définit l'application à utiliser pour traiter les messages de la file d'attente d'application. La définition de file d'attente d'application nomme le processus à utiliser et associe ainsi la file d'attente d'application à l'application à utiliser pour traiter ses messages. Cette opération est effectuée via l'attribut PROCESS de la file d'attente d'application MOTOR.INSURANCE.QUEUE. La commande MQSC suivante définit le processus requis, MOTOR.INSURANCE.QUOTE.PROCESS, identifié dans cet exemple:

```
DEFINE PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) +
DESCR ('Insurance request message processing') +
APPLTYPE (UNIX) +
APPLICID ('/u/admin/test/IRMP01') +
USERDATA ('open, close, 235')
```

où :

MOTOR.INSURANCE.QUOTE.PROCESS

Nom de la définition de processus.

DESCR ('Insurance request message processing')

Décrit le programme d'application auquel cette définition se rapporte. Ce texte s'affiche lorsque vous utilisez la commande DISPLAY PROCESS. Cela peut vous aider à identifier ce que le processus fait. Si vous utilisez des espaces dans la chaîne, vous devez placer la chaîne entre apostrophes.

APPLTYPE (UNIX)

Type d'application à démarrer.

APPLICID ('/u/admin/test/IRMP01')

Nom du fichier exécutable de l'application, indiqué comme nom de fichier qualifié complet. Sur les systèmes Windows, une valeur APPLICID typique serait c:\appl\test\irmp01.exe.

USERDATA ('open, close, 235')

Données définies par l'utilisateur qui peuvent être utilisées par l'application.

Affichage des attributs d'une définition de processus

Utilisez la commande DISPLAY PROCESS pour examiner les résultats de votre définition. Exemple :

```
DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)

24 : DISPLAY PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS) ALL
AMQ8407: Display Process details.
DESCR ('Insurance request message processing')
APPLICID ('/u/admin/test/IRMP01')
USERDATA (open, close, 235)
PROCESS (MOTOR.INSURANCE.QUOTE.PROCESS)
APPLTYPE (UNIX)
```

Vous pouvez également utiliser la commande MQSC ALTER PROCESS pour modifier une définition de processus existante et la commande DELETE PROCESS pour supprimer une définition de processus.

Utilisation de l'utilitaire dmpmqmsg entre deux systèmes

L'utilitaire **dmpmqmsg** (anciennement **qload**) est intégré au produit à partir de IBM MQ 8.0. Auparavant, l'utilitaire **qload** était disponible en tant que SupportPac MO03.

Présentation

L'utilitaire **dmpmqmsg** vous permet de copier ou de déplacer le contenu d'une file d'attente, ou ses messages, dans un fichier. Ce fichier peut être sauvegardé selon les besoins et utilisé ultérieurement pour recharger les messages dans la file d'attente.

Important : Le fichier possède un format spécifique compris par l'utilitaire. Toutefois, le fichier est lisible par l'utilisateur, de sorte que vous pouvez le mettre à jour dans un éditeur avant de le recharger. Si vous éditez le fichier, vous ne devez pas modifier son format.

Les utilisations possibles sont les suivantes:

- Enregistrement des messages d'une file d'attente dans un fichier. Possibilité à des fins d'archivage, puis rechargement dans une file d'attente.
- Rechargement d'une file d'attente avec les messages précédemment sauvegardés dans un fichier.
- Suppression des anciens messages d'une file d'attente.
- "Réexécution" des messages de test à partir d'un emplacement enregistré, en conservant même l'heure correcte entre les messages, le cas échéant.



Avertissement : SupportPac MO03 a utilisé le paramètre **-l** pour spécifier la liaison locale ou client. **-l** a été remplacé par le paramètre **-c**.

-P est désormais utilisé pour les informations de page de codes à la place de **-c**.

Pour plus d'informations sur la commande et les paramètres disponibles, voir [dmpmqmsg](#).

Exemple d'utilisation de l'utilitaire **dmpmqmsg** sous Linux, à l'aide d'une machine Windows

Vous disposez d'un gestionnaire de files d'attente sur une machine Linux comportant des messages dans une file d'attente (*Q1*) que vous souhaitez déplacer dans une autre file d'attente (*Q2*) dans le même gestionnaire de files d'attente. Vous souhaitez lancer l'utilitaire **dmpmqmsg** à partir d'une machine Windows.

La file d'attente (*Q1*) comporte quatre messages qui ont été ajoutés à l'aide de l'exemple d'application **amqspu**t (gestionnaire de files d'attente locales) ou **amqspu**tc (gestionnaire de files d'attente éloignées).

Sur la machine Linux, vous voyez:

```
display q1(Q1) CURDEPTH
      2 : display q1(Q1) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q1)
      TYPE(QLOCAL)
      CURDEPTH(4)
```

Définissez la variable d'environnement MQSERVER pour qu'elle pointe vers le gestionnaire de files d'attente dans Linux. Exemple :

```
set MQSERVER=SYSTEM.DEF.SVRCONN/TCP/veracruz.x.com(1414)
```

où *veracruz* est le nom de la machine.

Exécutez l'utilitaire **dmpmqmsg** pour lire la file d'attente, *Q1*, et stocker la sortie dans `c:\temp\mqqload.txt`.

Connectez-vous en tant que client distant au gestionnaire de files d'attente, *QM_VER*, s'exécutant sur l'hôte Linux et le port établis par MQSERVER. Vous obtenez la connexion en tant que client distant à l'aide de l'attribut: `-c`.

```
dmpmqmsg -m QM_VER -i Q1 -f c:\temp\mqqload.txt -c
Read    - Files:    0  Messages:    4  Bytes:    22
Written - Files:    1  Messages:    4  Bytes:    22
```

Le fichier de sortie `c:\temp\mqqload.txt` contient du texte, en utilisant un format compris par l'utilitaire **dmpmqmsg**.

Sur la machine Windows, exécutez la commande **dmpmqmsg** (à l'aide de l'option `-o` à la place de l'option `-i`) pour charger la file d'attente (*Q2*) sur la machine Linux à partir d'un fichier sur la machine Windows :

```
dmpmqmsg -m QM_VER -o Q2 -f c:\temp\mqqload.txt -c
Read    - Files:    1  Messages:    4  Bytes:    22
Written - Files:    0  Messages:    4  Bytes:    22
```

Sur la machine Linux, notez que quatre messages de la file d'attente ont été restaurés à partir du fichier.

```
display q1(Q2) CURDEPTH
      6 : display q1(Q2) CURDEPTH
AMQ8409: Display Queue details.
      QUEUE(Q2)
      TYPE(QLOCAL)
      CURDEPTH(4)
```

Sur la machine Linux,

Supprimez les messages de la file d'attente d'origine.

```
clear qlocal(Q1)
    4 : clear qlocal(Q1)
AMQ8022: IBM MQ queue cleared.
```

Confirmez qu'il n'y a plus de messages dans la file d'attente d'origine:

```
display ql(Q1) CURDEPTH
    5 : display ql(Q1) CURDEPTH
AMQ8409: Display Queue details.
    QUEUE(Q1)
    TYPE(QLLOCAL)
    CURDEPTH(0)
```

Pour obtenir une description de la commande et de ses paramètres, voir [dmpmqmsg](#) .

Concepts associés

[«Exemples d'utilisation de l'utilitaire dmpmqmsg», à la page 197](#)

Méthodes simples d'utilisation de l'utilitaire **dmpmqmsg** (anciennement **qload**). Cet utilitaire est intégré au produit à partir de IBM MQ 8.0.

Exemples d'utilisation de l'utilitaire dmpmqmsg

Méthodes simples d'utilisation de l'utilitaire **dmpmqmsg** (anciennement **qload**). Cet utilitaire est intégré au produit à partir de IBM MQ 8.0.

Auparavant, l'utilitaire **qload** était disponible en tant que SupportPac MO03.

Déchargement d'une file d'attente dans un fichier

Utilisez les options suivantes sur la ligne de commande pour sauvegarder les messages qui se trouvent dans une file d'attente dans un fichier:

```
dmpmqmsg -m QM1 -i Q1 -f c:\myfile
```

Cette commande prend une copie des messages de la file d'attente et les sauvegarde dans le fichier indiqué.

Déchargement d'une file d'attente dans une série de fichiers

Vous pouvez décharger une file d'attente dans une série de fichiers en utilisant le caractère `insert` dans le nom de fichier. Dans ce mode, chaque message est écrit dans un nouveau fichier:

```
dmpmqmsg -m QM1 -i Q1 -f c:\myfile%n
```

Cette commande décharge la file d'attente dans les fichiers `myfile1`, `myfile2`, `myfile3`, etc.

Chargement d'une file d'attente à partir d'un fichier

Pour recharger une file d'attente avec les messages que vous avez sauvegardés dans [«Déchargement d'une file d'attente dans un fichier», à la page 197](#), utilisez les options suivantes sur la ligne de commande:

```
dmpmqmsg -m QM1 -o Q1 -f c:\myfile%n
```

Cette commande décharge la file d'attente dans les fichiers `myfile1`, `myfile2`, `myfile3`, etc.

Chargement d'une file d'attente à partir d'une série de fichiers

Vous pouvez charger une file d'attente à partir d'une série de fichiers en utilisant le caractère `insert` dans le nom de fichier. Dans ce mode, chaque message est écrit dans un nouveau fichier:

```
dmpmqmsg -m QM1 -o Q1 -f c:\myfile%n
```

Cette commande charge la file d'attente dans les fichiers `myfile1`, `myfile2`, `myfile3`, etc.

Copier les messages d'une file d'attente vers une autre file d'attente

Remplacez le paramètre de fichier dans [«Déchargement d'une file d'attente dans un fichier»](#), à la page 197 par un autre nom de file d'attente et utilisez les options suivantes:

```
dmpmqmsg -m QM1 -i Q1 -o Q2
```

Cette commande permet de copier les messages d'une file d'attente vers une autre.

Copier les 100 premiers messages d'une file d'attente vers une autre

Utilisez la commande de l'exemple précédent et ajoutez l'option `-r#100` :

```
dmpmqmsg -m QM1 -i Q1 -o Q2 -r#100
```

Déplacer les messages d'une file d'attente vers une autre file d'attente

Une variante sur [«Chargement d'une file d'attente à partir d'un fichier»](#), à la page 197. Notez la distinction entre l'utilisation de `-i` (minuscules) qui ne parcourt qu'une file d'attente et l'utilisation de `-I` (majuscules) qui extrait une file d'attente de façon destructive:

```
dmpmqmsg -m QM1 -I Q1 -o Q2
```

Déplacer les messages de plus d'un jour d'une file d'attente vers une autre file d'attente

Cet exemple illustre l'utilisation de la sélection d'âge. Vous pouvez sélectionner des messages plus anciens, plus jeunes ou compris dans une plage d'âges.

```
dmpmqmsg -m QM1 -I Q1 -o Q2 -T1440
```

Afficher les âges des messages actuellement dans une file d'attente

Utilisez les options suivantes sur la ligne de commande:

```
dmpmqmsg -m QM1 -i Q1 -f stdout -dT
```

Gestion du fichier de messages

Après avoir déchargé le message de votre file d'attente, comme dans [«Déchargement d'une file d'attente dans un fichier»](#), à la page 197, vous pouvez éditer le fichier.

Vous pouvez également modifier le format du fichier afin d'utiliser l'une des options d'affichage que vous n'avez pas indiquées lors du déchargement de la file d'attente.

Vous pouvez utiliser l'utilitaire **dmpmqmsg** pour retraiter le fichier dans le format requis, même après le déchargement de la file d'attente. Utilisez les options suivantes sur la ligne de commande.

```
dmpmqmsg -f c:\oldfile -f c:\newfile -dA
```

Pour obtenir une description de la commande et de ses paramètres, voir [dmpmqmsg](#) .

Administration des objets IBM MQ éloignés

Cette section explique comment administrer des objets IBM MQ sur un gestionnaire de files d'attente éloignées à l'aide de commandes MQSC et comment utiliser des objets de file d'attente éloignée pour contrôler la destination des messages et des messages de réponse.

Procédure

- Pour plus d'informations sur l'administration des objets IBM MQ distants, voir les sous-rubriques suivantes:
 - [«Canaux et mise en file d'attente à distance»](#), à la page 199
 - [«Administration à distance à partir d'un gestionnaire de files d'attente local»](#), à la page 201
 - [«Création d'une définition locale d'une file d'attente éloignée»](#), à la page 207
 - [«Vérification de la fin des commandes asynchrones pour les réseaux distribués»](#), à la page 210
 - [«Utilisation de définitions de file d'attente éloignée comme alias»](#), à la page 212
 - [«Conversion de données»](#), à la page 213

Canaux et mise en file d'attente à distance

Vous pouvez configurer un canal pour fournir une liaison de communication unidirectionnelle entre un gestionnaire de files d'attente local et un gestionnaire de files d'attente éloignées. Le canal peut transmettre des messages destinés à n'importe quel nombre de files d'attente sur le gestionnaire de files d'attente éloignées. Si vous souhaitez que le gestionnaire de files d'attente éloignées réponde aux messages envoyés par le gestionnaire de files d'attente local, vous pouvez configurer un deuxième canal pour renvoyer les réponses du gestionnaire de files d'attente éloignées au gestionnaire de files d'attente local.

Un gestionnaire de files d'attente communique avec un autre gestionnaire de files d'attente en envoyant un message et, si nécessaire, en recevant une réponse. Le gestionnaire de files d'attente de réception peut être:

- Sur la même machine
- Sur une autre machine au même endroit (ou même de l'autre côté du monde)
- Exécution sur la même plateforme que le gestionnaire de files d'attente local
- Exécution sur une autre plateforme prise en charge par IBM MQ

Ces messages peuvent provenir de:

- Programmes d'application écrits par l'utilisateur qui transfèrent des données d'un noeud à un autre
- Applications d'administration écrites par l'utilisateur qui utilisent des commandes PCF ou MQAI
- IBM MQ Explorer,
- Envoi des gestionnaires de files d'attente:
 - Messages d'événement d'instrumentation vers un autre gestionnaire de files d'attente
 - Commandes MQSC émises à partir d'une commande **runmqsc** en mode indirect (où les commandes sont exécutées sur un autre gestionnaire de files d'attente)

Avant qu'un message puisse être envoyé à un gestionnaire de files d'attente éloignées, le gestionnaire de files d'attente local a besoin d'un mécanisme permettant de détecter l'arrivée des messages et de les transporter, à savoir:

- Au moins un canal
- Une file d'attente de transmission
- Un initiateur de canal

Pour qu'un gestionnaire de files d'attente éloignées reçoive un message, un programme d'écoute est requis.

Un canal est une liaison de communication unidirectionnelle entre deux gestionnaires de files d'attente et peut transporter des messages destinés à n'importe quel nombre de files d'attente sur le gestionnaire de files d'attente éloignées.

Chaque extrémité du canal possède une définition distincte. Par exemple, si une extrémité est un expéditeur ou un serveur, l'autre extrémité doit être un destinataire ou un demandeur. Un canal simple se compose d'une *définition de canal émetteur* à l'extrémité du gestionnaire de files d'attente local et d'une *définition de canal récepteur* à l'extrémité du gestionnaire de files d'attente éloigné. Les deux définitions doivent avoir le même nom et constituer ensemble un canal de transmission de messages unique.

Si vous souhaitez que le gestionnaire de files d'attente éloignées réponde aux messages envoyés par le gestionnaire de files d'attente local, configurez un deuxième canal pour renvoyer les réponses au gestionnaire de files d'attente local.

Utilisez la commande MQSC **DEFINE CHANNEL** pour définir des canaux. Dans cette section, les exemples relatifs aux canaux utilisent les attributs de canal par défaut, sauf indication contraire.

Il existe un agent MCA à chaque extrémité d'un canal, qui contrôle l'envoi et la réception des messages. L'agent MCA extrait les messages de la file d'attente de transmission et les place sur la liaison de communication entre les gestionnaires de files d'attente.

Une file d'attente de transmission est une file d'attente locale spécialisée qui contient temporairement des messages avant que l'agent MCA ne les récupère et les envoie au gestionnaire de files d'attente éloignées. Vous spécifiez le nom de la file d'attente de transmission dans une *définition de file d'attente éloignée*.

Vous pouvez autoriser un agent MCA à transférer des messages via plusieurs unités d'exécution. Ce processus est connu sous le nom de *principe du pipeline*. Le principe du pipeline permet à l'agent MCA de transférer des messages plus efficacement, améliorant ainsi les performances de canal. Pour plus d'informations sur la configuration d'un canal en vue de l'utilisation du pipeline, voir [Attributs des canaux](#).

«Préparation des canaux et des files d'attente de transmission pour l'administration à distance», à la page [202](#) vous explique comment utiliser ces définitions pour configurer l'administration à distance.

Pour plus d'informations sur la configuration de la mise en file d'attente répartie en général, voir [Composants de la mise en file d'attente répartie](#).

Information associée

[runmqsc](#) (exécution des commandes MQSC)

Administration à distance à l'aide de clusters

Dans un réseau IBM MQ utilisant la mise en file d'attente répartie, chaque gestionnaire de files d'attente est indépendant. Si un gestionnaire de files d'attente doit envoyer des messages à un autre gestionnaire de files d'attente, il doit définir une file d'attente de transmission, un canal vers le gestionnaire de files d'attente éloignées et une définition de file d'attente éloignée pour chaque file d'attente à laquelle il souhaite envoyer des messages.

Un *cluster* est un groupe de gestionnaires de files d'attente configurés de telle sorte que les gestionnaires de files d'attente puissent communiquer directement entre eux sur un réseau unique sans file d'attente de transmission, canal et définitions de file d'attente complexes. Les clusters peuvent être configurés facilement et contiennent généralement des gestionnaires de files d'attente qui sont logiquement liés

d'une manière ou d'une autre et qui doivent partager des données ou des applications. Même le plus petit cluster réduit les coûts d'administration du système.

L'établissement d'un réseau de gestionnaires de files d'attente dans un cluster implique moins de définitions que l'établissement d'un environnement de mise en file d'attente réparti traditionnel. Avec moins de définitions à créer, vous pouvez configurer ou modifier votre réseau plus rapidement et plus facilement, et réduire le risque d'erreur dans vos définitions.






Pour configurer un cluster, vous avez besoin d'une définition d'émetteur de cluster (CLUSDR) et d'un récepteur de cluster (CLUSRCVR) pour chaque gestionnaire de files d'attente. Vous n'avez pas besoin de définitions de file d'attente de transmission ni de définitions de file d'attente éloignée. Les principes de l'administration à distance sont les mêmes lorsqu'ils sont utilisés dans un cluster, mais les définitions elles-mêmes sont grandement simplifiées.

Administration à distance à partir d'un gestionnaire de files d'attente local

Vous pouvez administrer un gestionnaire de files d'attente éloignées à partir d'un gestionnaire de files d'attente local à l'aide des commandes MQSC et PCF.

La préparation des files d'attente et des canaux est essentiellement la même pour les commandes MQSC et PCF. Dans cette section, les exemples montrent des commandes MQSC, car elles sont plus faciles à comprendre. Pour plus d'informations sur l'écriture de programmes d'administration à l'aide de commandes PCF, voir [«Utilisation des formats de commande programmables IBM MQ»](#), à la page 23.

Vous envoyez des commandes MQSC à un gestionnaire de files d'attente éloignées de manière interactive ou à partir d'un fichier texte contenant les commandes. Le gestionnaire de files d'attente éloignées peut se trouver sur la même machine ou, plus généralement, sur une machine différente. Vous pouvez administrer à distance des gestionnaires de files d'attente dans d'autres environnements IBM MQ, notamment:

-  UNIX
-  Linux
-  Windows
-  IBM i
-  z/OS

Pour implémenter l'administration à distance, vous devez créer des objets spécifiques. A moins que vous ne soyez soumis à des exigences spécifiques, les valeurs par défaut, par exemple, pour la longueur maximale des messages, sont suffisantes.

Préparation des gestionnaires de files d'attente pour l'administration à distance

Comment utiliser les commandes MQSC pour préparer les gestionnaires de files d'attente pour l'administration à distance.

La [Figure 18](#), à la page 202 présente la configuration des gestionnaires de files d'attente et des canaux dont vous avez besoin pour l'administration à distance à l'aide de la commande **runmqsc**. L'objet `source.queue.manager` est le gestionnaire de files d'attente source à partir duquel vous pouvez émettre des commandes MQSC et auquel les résultats de ces commandes (messages de l'opérateur) sont renvoyés. L'objet `target.queue.manager` est le nom du gestionnaire de files d'attente cible, qui traite les commandes et génère des messages d'opérateur.

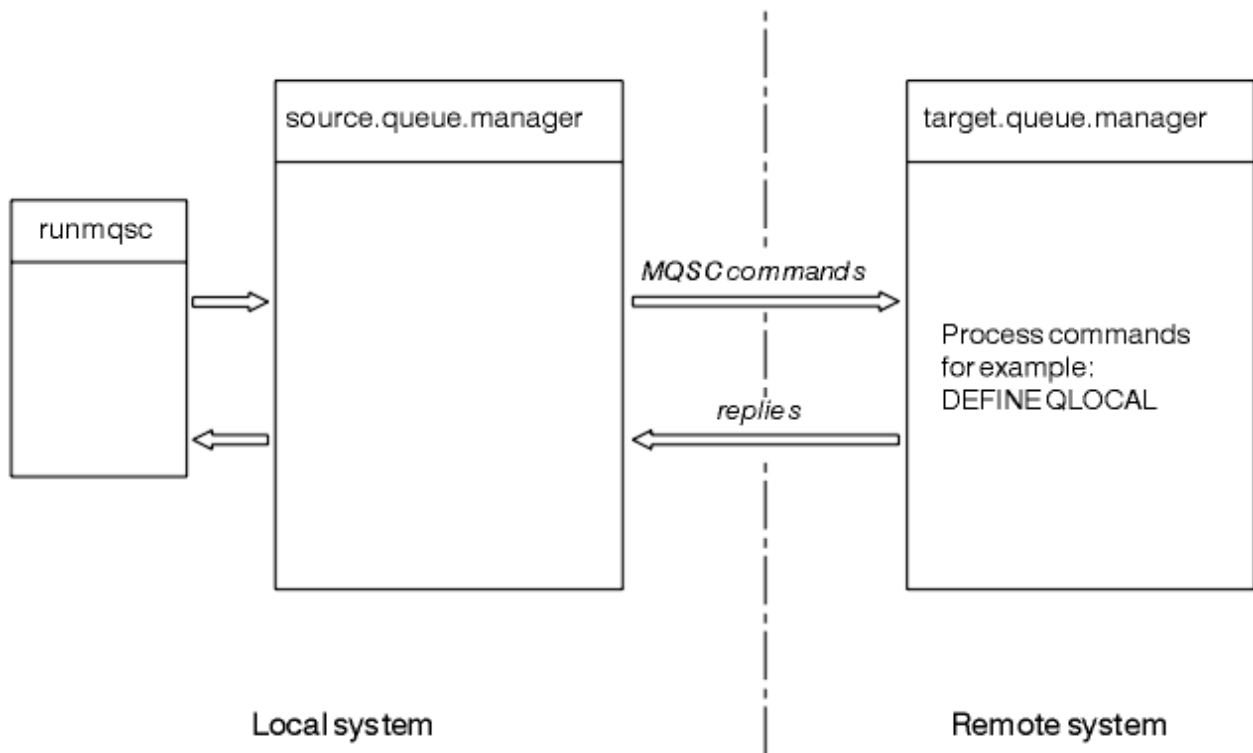


Figure 18. Administration à distance à l'aide des commandes MQSC

Sur les deux systèmes, si vous ne l'avez pas déjà fait:

- Créez le gestionnaire de files d'attente et les objets par défaut à l'aide de la commande `crtmqm`. Pour plus d'informations, voir [crtmqm](#).
- Démarrez le gestionnaire de files d'attente à l'aide de la commande `strmqm`. Pour plus d'informations, voir [strmqm](#).

Sur le gestionnaire de files d'attente cible:

- La file d'attente de commandes, `SYSTEM.ADMIN.COMMAND.QUEUE`, doit être présente. Cette file d'attente est créée par défaut lorsqu'un gestionnaire de files d'attente est créé.

Vous devez exécuter ces commandes localement ou via une fonction réseau telle que Telnet.

Préparation des canaux et des files d'attente de transmission pour l'administration à distance

Comment utiliser les commandes MQSC pour préparer les canaux et les files d'attente de transmission pour l'administration à distance.

Pour exécuter des commandes MQSC à distance, configurez deux canaux, un pour chaque direction et leurs files d'attente de transmission associées. Cet exemple suppose que vous utilisez TCP/IP comme type de transport et que vous connaissez l'adresse TCP/IP impliquée.

Le canal `source.to.target` permet d'envoyer des commandes MQSC du gestionnaire de files d'attente source vers le gestionnaire de files d'attente cible. Son émetteur est `source.queue.manager` et son récepteur est `target.queue.manager`. Le canal `target.to.source` permet de renvoyer la sortie des commandes et des messages de l'opérateur générés vers le gestionnaire de files d'attente source. Vous devez également définir une file d'attente de transmission pour chaque canal. Cette file d'attente est une file d'attente locale à laquelle est attribué le nom du gestionnaire de files d'attente de réception. Le nom `XMITQ` doit correspondre au nom du gestionnaire de files d'attente éloignées pour que l'administration à distance fonctionne, sauf si vous utilisez un alias de gestionnaire de files d'attente. Le [Figure 19](#), à la page [203](#) récapitule cette configuration.

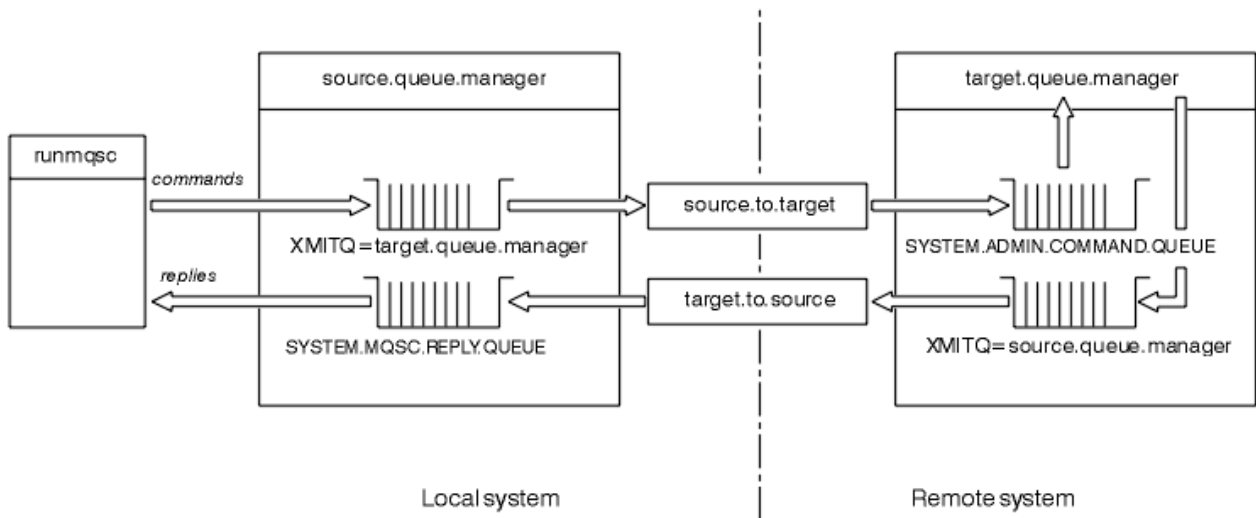


Figure 19. Configuration de canaux et de files d'attente pour l'administration à distance

Pour plus d'informations sur la configuration des canaux, voir [Configuration de la mise en file d'attente répartie](#).

Définition des canaux, des programmes d'écoute et des files d'attente de transmission

Sur le gestionnaire de files d'attente source (`source.queue.manager`), exécutez les commandes MQSC suivantes pour définir les canaux, le programme d'écoute et la file d'attente de transmission:

1. Définissez le canal émetteur sur le gestionnaire de files d'attente source:

```
DEFINE CHANNEL ('source.to.target') +
CHLTYPE(SDR) +
CONNNAME (RHX5498) +
XMITQ ('target.queue.manager') +
TRPTYPE(TCP)
```

2. Définissez le canal récepteur sur le gestionnaire de files d'attente source:

```
DEFINE CHANNEL ('target.to.source') +
CHLTYPE(RCVR) +
TRPTYPE(TCP)
```

3. Définissez le programme d'écoute sur le gestionnaire de files d'attente source:

```
DEFINE LISTENER ('source.queue.manager') +
TRPTYPE (TCP)
```

4. Définissez la file d'attente de transmission sur le gestionnaire de files d'attente source:

```
DEFINE QLOCAL ('target.queue.manager') +
USAGE (XMITQ)
```

Exécutez les commandes suivantes sur le gestionnaire de files d'attente cible (`target.queue.manager`) pour créer les canaux, le programme d'écoute et la file d'attente de transmission:

1. Définissez le canal émetteur sur le gestionnaire de files d'attente cible:

```
DEFINE CHANNEL ('target.to.source') +
CHLTYPE(SDR) +
CONNNAME (RHX7721) +
```

```
XMITQ ('source.queue.manager') +  
TRPTYPE(TCP)
```

2. Définissez le canal récepteur sur le gestionnaire de files d'attente cible:

```
DEFINE CHANNEL ('source.to.target') +  
CHLTYPE(RCVR) +  
TRPTYPE(TCP)
```

3. Définissez le programme d'écoute sur le gestionnaire de files d'attente cible:

```
DEFINE LISTENER ('target.queue.manager') +  
TRPTYPE (TCP)
```

4. Définissez la file d'attente de transmission sur le gestionnaire de files d'attente cible:

```
DEFINE QLOCAL ('source.queue.manager') +  
USAGE (XMITQ)
```

Remarque : Les noms de connexion TCP/IP indiqués pour l'attribut CONNAME dans les définitions de canal émetteur ne sont donnés qu'à titre d'illustration. Il s'agit du nom de réseau de la machine à l' *autre* extrémité de la connexion. Utilisez les valeurs appropriées pour votre réseau.

Démarrage des programmes d'écoute et des canaux

Comment utiliser les commandes MQSC pour démarrer les programmes d'écoute et les canaux.

Démarrez les deux programmes d'écoute à l'aide des commandes MQSC suivantes:

1. Démarrez le programme d'écoute sur le gestionnaire de files d'attente source, `source.queue.manager`, en exécutant la commande MQSC suivante:

```
START LISTENER ('source.queue.manager')
```

2. Démarrez le programme d'écoute sur le gestionnaire de files d'attente cible, `target.queue.manager`, en exécutant la commande MQSC suivante:

```
START LISTENER ('target.queue.manager')
```

Démarrez les deux canaux émetteurs à l'aide des commandes MQSC suivantes:

1. Démarrez le canal émetteur sur le gestionnaire de files d'attente source, `source.queue.manager`, en émettant la commande MQSC suivante:

```
START CHANNEL ('source.to.target')
```

2. Démarrez le canal émetteur sur le gestionnaire de files d'attente cible, `target.queue.manager`, en exécutant la commande MQSC suivante:

```
START CHANNEL ('target.to.source')
```

Définition automatique des canaux

Vous activez la définition automatique des définitions de récepteur et de connexion serveur en mettant à jour l'objet gestionnaire de files d'attente à l'aide de la commande MQSC ALTER QMGR (ou de la commande PCF Change Queue Manager).

Si IBM MQ reçoit une demande d'association entrante et ne trouve pas de canal de réception ou de connexion serveur approprié, il crée automatiquement un canal. Les définitions automatiques sont basées sur deux définitions par défaut fournies avec IBM MQ: SYSTEM.AUTO.RECEIVER et SYSTEM.AUTO.SVRCONN.

Pour plus d'informations sur la création automatique de définitions de canal, voir [Préparation des canaux](#). Pour plus d'informations sur la définition automatique de canaux pour les clusters, voir [Utilisation de canaux définis automatiquement](#).

Gestion du serveur de commandes pour l'administration à distance

Comment démarrer, arrêter et afficher le statut du serveur de commandes. Un serveur de commandes est obligatoire pour toute administration impliquant des commandes PCF, MQAI, ainsi que pour l'administration à distance.

Chaque gestionnaire de files d'attente peut être associé à un serveur de commandes. Un serveur de commandes traite toutes les commandes entrantes provenant des gestionnaires de files d'attente éloignées ou les commandes PCF provenant des applications. Il présente les commandes au gestionnaire de files d'attente pour le traitement et renvoie un code achèvement ou un message d'opérateur en fonction de l'origine de la commande.

Remarque : Pour l'administration à distance, vérifiez que le gestionnaire de files d'attente cible est en cours d'exécution. Sinon, les messages contenant des commandes ne peuvent pas quitter le gestionnaire de files d'attente à partir duquel elles sont émises. A la place, ces messages sont mis en file d'attente dans la file d'attente de transmission locale qui sert le gestionnaire de files d'attente éloignées. Évitez cette situation.

Il existe des commandes de contrôle distinctes pour le démarrage et l'arrêt du serveur de commandes. A condition que le serveur de commandes soit en cours d'exécution, les utilisateurs de IBM MQ for Windows ou IBM MQ for Linux (plateformes x86 et x86-64) peuvent effectuer les opérations décrites dans les sections suivantes à l'aide de l'explorateur IBM MQ. Pour plus d'informations, voir «Administration à l'aide de IBM MQ Explorer», à la page 135.

Démarrage du serveur de commandes

En fonction de la valeur de l'attribut de gestionnaire de files d'attente, *SCMDSERV*, le serveur de commandes est démarré automatiquement au démarrage du gestionnaire de files d'attente ou doit être démarré manuellement. La valeur de l'attribut de gestionnaire de files d'attente peut être modifiée à l'aide de la commande MQSC ALTER QMGR spécifiant le paramètre SCMDSERV. Par défaut, le serveur de commandes est démarré automatiquement.

Si *SCMDSERV* est défini sur MANUAL, démarrez le serveur de commandes à l'aide de la commande suivante:

```
stimqcsv saturn.queue.manager
```

où *saturn.queue.manager* est le gestionnaire de files d'attente pour lequel le serveur de commandes est démarré.

Affichage de l'état du serveur de commandes

Pour l'administration à distance, vérifiez que le serveur de commandes sur le gestionnaire de files d'attente cible est en cours d'exécution. S'il n'est pas en cours d'exécution, les commandes distantes ne peuvent pas être traitées. Tous les messages contenant des commandes sont mis en file d'attente dans la file d'attente de commandes du gestionnaire de files d'attente cible.

Pour afficher le statut du serveur de commandes d'un gestionnaire de files d'attente, exécutez la commande MQSC suivante:

```
DISPLAY QMSTATUS CMDSERV
```

Arrêt d'un serveur de commandes

Pour arrêter le serveur de commandes démarré par l'exemple précédent, utilisez la commande suivante:

```
endmqcsv saturn.queue.manager
```

Vous pouvez arrêter le serveur de commandes de deux manières:

- Pour un arrêt contrôlé, utilisez la commande `endmqcsv` avec l'indicateur `-c`, qui est la valeur par défaut.
- Pour un arrêt immédiat, utilisez la commande `endmqcsv` avec l'indicateur `-i`.

Remarque : L'arrêt d'un gestionnaire de files d'attente arrête également le serveur de commandes qui lui est associé.

Emission de commandes MQSC sur un gestionnaire de files d'attente éloignées

Vous pouvez utiliser une forme particulière de la commande `runmqsc` pour exécuter des commandes MQSC sur un gestionnaire de files d'attente éloignées.

Le serveur de commandes doit être en cours d'exécution sur le gestionnaire de files d'attente cible, s'il va traiter des commandes MQSC à distance. (Cette opération n'est pas nécessaire sur le gestionnaire de files d'attente source). Pour plus d'informations sur le démarrage du serveur de commandes sur un gestionnaire de files d'attente, voir [«Gestion du serveur de commandes pour l'administration à distance»](#), à la page 205.

Sur le gestionnaire de files d'attente source, vous pouvez ensuite exécuter les commandes MQSC de manière interactive en mode indirect en entrant:

```
runmqsc -w 30 -m source.queue.manager target.queue.manager
```

Cette forme de la commande `runmqsc`, avec l'indicateur `-w`, exécute les commandes MQSC en mode indirect, où les commandes sont placées (sous une forme modifiée) dans la file d'attente d'entrée du serveur de commandes et exécutées dans l'ordre.

Lorsque vous entrez une commande MQSC, elle est redirigée vers le gestionnaire de files d'attente éloignées, dans ce cas, `target.queue.manager`. Le délai d'attente est de 30 secondes ; si une réponse n'est pas reçue dans les 30 secondes, le message suivant est généré sur le gestionnaire de files d'attente local (source):

```
AMQ8416: MQSC timed out waiting for a response from the command server.
```

Lorsque vous arrêtez d'émettre des commandes MQSC, le gestionnaire de files d'attente local affiche toutes les réponses arrivées à expiration et supprime toutes les autres réponses.

Le gestionnaire de files d'attente source prend par défaut la valeur du gestionnaire de files d'attente local par défaut. Si vous spécifiez l'option `-m LocalQmgrNom` dans la commande `runmqsc`, vous pouvez indiquer les commandes à émettre par l'intermédiaire d'un gestionnaire de files d'attente local.

En mode indirect, vous pouvez également exécuter un fichier de commandes MQSC sur un gestionnaire de files d'attente éloignées. Exemple :

```
runmqsc -w 60 target.queue.manager < mycomds.in > report.out
```

où `mycomds.in` est un fichier contenant les commandes MQSC et `report.out` est le fichier de rapport.

Méthode suggérée pour l'émission de commandes à distance

Lorsque vous émettez des commandes sur un gestionnaire de files d'attente éloignées, envisagez d'utiliser l'approche suivante:

1. Placez les commandes MQSC à exécuter sur le système distant dans un fichier de commandes.
2. Vérifiez vos commandes MQSC en local en spécifiant l'indicateur `-v` dans la commande `runmqsc`.

Vous ne pouvez pas utiliser `runmqsc` pour vérifier les commandes MQSC sur un autre gestionnaire de files d'attente.

3. Vérifiez que le fichier de commandes s'exécute localement sans erreur.
4. Exécutez le fichier de commandes sur le système distant.

Si vous rencontrez des problèmes lors de l'utilisation à distance des commandes MQSC

Si vous avez des difficultés à exécuter des commandes MQSC à distance, vérifiez que vous disposez des éléments suivants:

- Démarrage du serveur de commandes sur le gestionnaire de files d'attente cible.
- Une file d'attente de transmission valide a été définie.
- Définissez les deux extrémités des canaux de transmission de messages pour les deux:
 - Canal via lequel les commandes sont envoyées.
 - Canal par lequel les réponses doivent être renvoyées.
- Indique le nom de connexion correct (CONNNAME) dans la définition de canal.
- Vous avez démarré les programmes d'écoute avant de démarrer les canaux de transmission de messages.
- Vérification que l'intervalle de déconnexion n'a pas expiré, par exemple, si un canal a démarré, puis s'est arrêté après un certain temps. Ceci est particulièrement important si vous démarrez les canaux manuellement.
- Demandes envoyées à partir d'un gestionnaire de files d'attente source qui n'ont pas de sens pour le gestionnaire de files d'attente cible (par exemple, des demandes incluant des paramètres qui ne sont pas pris en charge sur le gestionnaire de files d'attente éloignées).

Voir aussi [«Résolution des problèmes liés aux commandes MQSC»](#), à la page 20.

Utilisation des gestionnaires de files d'attente sous z/OS

Vous pouvez émettre des commandes MQSC vers un gestionnaire de files d'attente z/OS à partir d'un gestionnaire de files d'attente sur les plateformes décrites dans ce guide. Toutefois, pour ce faire, vous devez modifier la commande `runmqsc` et les définitions de canal au niveau de l'émetteur.

En particulier, vous ajoutez l'indicateur `-x` à la commande `runmqsc` sur le noeud source pour indiquer que le gestionnaire de files d'attente cible est en cours d'exécution sous z/OS:

```
runmqsc -w 30 -x target.queue.manager
```

Création d'une définition locale d'une file d'attente éloignée

Une définition locale d'une file d'attente éloignée est une définition sur un gestionnaire de files d'attente local qui fait référence à une file d'attente sur un gestionnaire de files d'attente éloignées.

Vous n'avez pas besoin de définir une file d'attente éloignée à partir d'une position locale, mais l'avantage est que les applications peuvent faire référence à la file d'attente éloignée par son nom défini localement au lieu d'avoir à spécifier un nom qualifié par l'ID du gestionnaire de files d'attente sur lequel se trouve la file d'attente éloignée.

Description du fonctionnement des définitions locales des files d'attente distantes

Une application se connecte à un gestionnaire de files d'attente local, puis émet un appel MQOPEN . Dans l'appel ouvert, le nom de file d'attente spécifié est celui d'une définition de file d'attente éloignée sur le gestionnaire de files d'attente local. La définition de file d'attente éloignée fournit les noms de la file d'attente cible, du gestionnaire de files d'attente cible et, éventuellement, d'une file d'attente de transmission. Pour placer un message dans la file d'attente éloignée, l'application émet un appel MQPUT

en spécifiant le descripteur renvoyé par l'appel MQOPEN . Le gestionnaire de files d'attente utilise le nom de la file d'attente éloignée et le nom du gestionnaire de files d'attente éloignées dans un en-tête de transmission au début du message. Ces informations sont utilisées pour acheminer le message vers sa destination correcte dans le réseau.

En tant qu'administrateur, vous pouvez contrôler la destination du message en modifiant la définition de la file d'attente éloignée.

L'exemple suivant montre comment une application insère un message dans une file d'attente appartenant à un gestionnaire de files d'attente éloignées. L'application se connecte à un gestionnaire de files d'attente, par exemple, saturn.queue.manager. La file d'attente cible appartient à un autre gestionnaire de files d'attente.

Dans l'appel MQOPEN , l'application spécifie les zones suivantes:

| Zone Valeur | Description |
|--|---|
| <i>ObjectName</i> CYAN.REMOTE.QUEUE | Indique le nom local de l'objet file d'attente éloignée. Définit la file d'attente cible et le gestionnaire de files d'attente cible. |
| <i>ObjectType</i> (File d'attente) | Identifie cet objet en tant que file d'attente. |
| <i>ObjectQmgrName</i> Vide ou saturn.queue.manager | Cette zone est facultative. Si cette zone est vide, le nom du gestionnaire de files d'attente local est pris en compte. (Il s'agit du gestionnaire de files d'attente sur lequel la définition de file d'attente éloignée existe.) |

Après cela, l'application émet un appel MQPUT pour insérer un message dans cette file d'attente.

Sur le gestionnaire de files d'attente local, vous pouvez créer une définition locale d'une file d'attente éloignée à l'aide des commandes MQSC suivantes:

```
DEFINE QREMOTE (CYAN.REMOTE.QUEUE) +
DESCR ('Queue for auto insurance requests from the branches') +
RNAME (AUTOMOBILE.INSURANCE.QUOTE.QUEUE) +
RQMNAME (jupiter.queue.manager) +
XMITQ (INQUOTE.XMIT.QUEUE)
```

où :

QREMOTE (CYAN.REMOTE.QUEUE)

Indique le nom local de l'objet file d'attente éloignée. Il s'agit du nom que les applications connectées à ce gestionnaire de files d'attente doivent spécifier dans l'appel MQOPEN pour ouvrir la file d'attente AUTOMOBILE.INSURANCE.QUOTE.QUEUE sur le gestionnaire de files d'attente éloignées jupiter.queue.manager.

DESCR ('Queue for auto insurance requests from the branches')

Fournit un texte supplémentaire qui décrit l'utilisation de la file d'attente.

RNAME (AUTOMOBILE.INSURANCE.QUOTE.QUEUE)

Indique le nom de la file d'attente cible sur le gestionnaire de files d'attente éloignées. Il s'agit de la file d'attente cible réelle pour les messages envoyés par les applications qui spécifient le nom de file d'attente CYAN.REMOTE.QUEUE. File d'attente AUTOMOBILE.INSURANCE.QUOTE.QUEUE doit être définie en tant que file d'attente locale sur le gestionnaire de files d'attente éloignées.

RQMNAME (jupiter.queue.manager)

Indique le nom du gestionnaire de files d'attente éloignées qui possède la file d'attente cible AUTOMOBILE.INSURANCE.QUOTE.QUEUE.

XMITQ (INQUOTE.XMIT.QUEUE)

Définit le nom de la file d'attente de transmission. Cette option est facultative ; si le nom d'une file d'attente de transmission n'est pas spécifié, une file d'attente portant le même nom que le gestionnaire de files d'attente éloignées est utilisée.

Dans les deux cas, la file d'attente de transmission appropriée doit être définie en tant que file d'attente locale avec un attribut **Usage** spécifiant qu'il s'agit d'une file d'attente de transmission (USAGE (XMITQ)) dans les commandes MQSC).

Autre méthode d'insertion de messages dans une file d'attente éloignée

L'utilisation d'une définition locale d'une file d'attente éloignée n'est pas le seul moyen d'insérer des messages dans une file d'attente éloignée. Les applications peuvent spécifier le nom complet de la file d'attente, y compris le nom du gestionnaire de files d'attente éloignées, dans le cadre de l'appel MQOPEN . Dans ce cas, vous n'avez pas besoin d'une définition locale d'une file d'attente éloignée. Toutefois, cela signifie que les applications doivent connaître le nom du gestionnaire de files d'attente éloignées ou y avoir accès lors de l'exécution.

Utilisation d'autres commandes avec des files d'attente distantes

Vous pouvez utiliser les commandes MQSC pour afficher ou modifier les attributs d'un objet de file d'attente éloignée, ou vous pouvez supprimer l'objet de file d'attente éloignée. Exemple :

- Pour afficher les attributs de la file d'attente éloignée:

```
DISPLAY QUEUE (CYAN.REMOTE.QUEUE)
```

- Pour modifier la file d'attente éloignée afin d'activer les insertions. Cela n'affecte pas la file d'attente cible, mais uniquement les applications qui spécifient cette file d'attente éloignée:

```
ALTER QREMOTE (CYAN.REMOTE.QUEUE) PUT(ENABLED)
```

- Permet de supprimer cette file d'attente éloignée. Cela n'affecte pas la file d'attente cible, mais uniquement sa définition locale:

```
DELETE QREMOTE (CYAN.REMOTE.QUEUE)
```

Remarque : Lorsque vous supprimez une file d'attente éloignée, vous supprimez uniquement la représentation locale de la file d'attente éloignée. Vous ne supprimez pas la file d'attente éloignée elle-même ou les messages qu'elle contient.

Définition d'une file d'attente de transmission

Une file d'attente de transmission est une file d'attente locale utilisée lorsqu'un gestionnaire de files d'attente transfère des messages à un gestionnaire de files d'attente éloignées via un canal de transmission.

Le canal fournit une liaison unidirectionnelle vers le gestionnaire de files d'attente éloignées. Les messages sont placés dans la file d'attente de transmission jusqu'à ce que le canal puisse les accepter. Lorsque vous définissez un canal, vous devez spécifier un nom de file d'attente de transmission à l'extrémité émettrice du canal de transmission.

L'attribut de commande MQSC USAGE définit si une file d'attente est une file d'attente de transmission ou une file d'attente normale.

Files d'attente de transmission par défaut

Lorsqu'un gestionnaire de files d'attente envoie des messages à un gestionnaire de files d'attente éloignées, il identifie la file d'attente de transmission dans l'ordre suivant :

1. La file d'attente de transmission nommée sur l'attribut XMITQ de la définition locale d'une file d'attente éloignée.
2. Une file d'attente de transmission portant le même nom que le gestionnaire de files d'attente cible. (Cette valeur est la valeur par défaut sur XMITQ de la définition locale d'une file d'attente éloignée.)
3. La file d'attente de transmission nommée sur l'attribut DEFXMITQ du gestionnaire de files d'attente local.

Par exemple, la commande MQSC suivante crée une file d'attente de transmission par défaut sur source.queue.manager pour les messages destinés à target.queue.manager:

```
DEFINE QLOCAL ('target.queue.manager') +
DESCR ('Default transmission queue for target qm') +
USAGE (XMITQ)
```

Les applications peuvent placer les messages directement dans une file d'attente de transmission ou indirectement via une définition de file d'attente éloignée. Voir aussi [«Création d'une définition locale d'une file d'attente éloignée»](#), à la page 207.

Vérification de la fin des commandes asynchrones pour les réseaux distribués

De nombreuses commandes sont asynchrones lorsqu'elles sont utilisées dans un réseau réparti. En fonction de la commande et de l'état du réseau lorsqu'elle est émise, son exécution peut prendre beaucoup de temps. Le gestionnaire de files d'attente n'émet pas de message à la fin de l'exécution. Vous avez donc besoin d'autres méthodes pour vérifier que la commande est terminée.

Pourquoi et quand exécuter cette tâche

Presque tous les changements de configuration que vous apportez à un cluster sont susceptibles de se terminer de manière asynchrone. Cela est dû à l'administration interne et aux cycles de mise à jour qui fonctionnent au sein des clusters. Pour les hiérarchies de publication / abonnement, tout changement de configuration affectant les abonnements risque de se terminer de manière asynchrone. Cela n'est pas toujours évident à partir du nom de la commande.

Les commandes MQSC suivantes peuvent toutes être exécutées de manière asynchrone. Chacune de ces commandes possède un équivalent PCF et la plupart sont également disponibles dans IBM MQ Explorer . Lorsqu'elles sont exécutées sur un petit réseau sans charge de travail, ces commandes s'exécutent généralement en quelques secondes. Cependant, ce n'est pas le cas pour les réseaux plus grands et plus occupés. En outre, la commande **REFRESH CLUSTER** peut prendre beaucoup plus de temps, en particulier lorsqu'elle est émise sur plusieurs gestionnaires de files d'attente en même temps.

Pour être sûr que ces commandes sont terminées, vérifiez que les objets attendus existent sur les gestionnaires de files d'attente éloignées.

Procédure

- ALTER QMGR

Pour la commande ALTER QMGR PARENT , utilisez `DISPLAY PUBSUB TYPE(PARENT) ALL` pour suivre le statut de la relation parent demandée.

Pour les commandes ALTER QMGR REPOS et ALTER QMGR REPOSNL , utilisez `DISPLAY CLUSQMGR QMTYPE` pour confirmer l'exécution.

- DEFINE CHANNEL, ALTER CHANNEL et DELETE CHANNEL

Pour tous les paramètres répertoriés dans le tableau ALTER CHANNEL parameters, utilisez la commande `DISPLAY CLUSQMGR` pour surveiller le moment où les modifications ont été propagées au cluster.

- DEFINE NAMELIST, ALTER NAMELIST et DELETE NAMELIST.

Si vous utilisez un **NAMELIST** sur l'attribut **CLUSNL** d'un objet **QMgr**, une file d'attente ou un canal de cluster peut affecter cet objet. Surveillez l'objet concerné selon les besoins.

Les modifications apportées à **SYSTEM.QPUBSUB.QUEUE.NAMELIST** peuvent affecter la création ou l'annulation d'abonnements de proxy dans une hiérarchie de publication / abonnement. Utilisez la commande **DISPLAY SUB SUBTYPE (PROXY)** pour la surveiller.

- **DEFINE queues**, **ALTER queues** et **DELETE queues**.

Pour tous les paramètres répertoriés dans le tableau Paramètres pouvant être renvoyés par la commande DISPLAY QUEUE, utilisez la commande **DISPLAY QCLUSTER** pour surveiller le moment où les modifications ont été propagées au cluster.

- **DEFINE SUB** et **DELETE SUB**

Lorsque vous définissez le premier abonnement sur une chaîne de rubrique, vous pouvez créer des abonnements de proxy dans une hiérarchie de publication / abonnement ou dans un cluster de publication / abonnement. De même, lorsque vous supprimez le dernier abonnement sur une chaîne de rubrique, vous pouvez annuler des abonnements de proxy dans une hiérarchie de publication / abonnement ou dans un cluster de publication / abonnement.

Pour vérifier qu'une commande de définition ou de suppression d'un abonnement est terminée, vérifiez si l'abonnement de proxy attendu existe sur d'autres gestionnaires de files d'attente du réseau distribué. Si vous utilisez le *rou tage direct* dans un cluster, vérifiez que l'abonnement de proxy attendu existe sur les autres référentiels partiels du cluster. Si vous utilisez le *rou tage de l'hôte de rubrique* dans un cluster, vérifiez que l'abonnement de proxy attendu existe sur les hôtes de rubrique correspondants. Utilisez la commande MQSC suivante:

```
DISPLAY SUB(*) SUBTYPE(PROXY)
```

Utilisez la même vérification pour les appels MQI d'abonnement et de désabonnement équivalents suivants, lorsqu'ils sont émis dans un cluster ou une hiérarchie:

- Abonnez-vous à l'aide de **MQSUB**.
- Désabonnez-vous en utilisant **MQCLOSE** avec **MQCO_REMOVE_SUB**.

- **DEFINE TOPIC**, **ALTER TOPIC** et **DELETE TOPIC**

Pour vérifier qu'une commande de définition, de modification ou de suppression d'une rubrique en cluster est terminée, affichez la rubrique dans les autres référentiels partiels du cluster (si vous utilisez le *rou tage direct*) ou sur les autres hôtes de rubrique (si vous utilisez le *rou tage via un hôte de rubrique*).

Pour tous les paramètres répertoriés dans le tableau Paramètres pouvant être renvoyés par la commande DISPLAY TOPIC, utilisez la commande **DISPLAY TCLUSTER** pour surveiller le moment où les modifications ont été propagées au cluster.

Remarque :

- Le paramètre **CLUSTER** peut affecter la création ou l'annulation d'abonnements de proxy dans un cluster de publication / abonnement.
- Les paramètres **PROXYSUB** et **SUBSCOPE** peuvent affecter la création ou l'annulation d'abonnements de proxy dans une hiérarchie de publication / abonnement ou dans un cluster de publication / abonnement.
- Utilisez la commande **DISPLAY SUB SUBTYPE (PROXYSUB)** pour la surveiller.

- Actualiser le cluster

Si vous exécutez la commande **REFRESH CLUSTER**, interrogez la longueur de la file d'attente de commandes du cluster. Attendez qu'il atteigne zéro, et restez à zéro, avant de rechercher les objets.

1. Utilisez la commande MQSC suivante pour vérifier que la longueur de la file d'attente de commandes du cluster est égale à zéro.

```
DISPLAY QL(SYSTEM.CLUSTER.COMMAND.QUEUE) CURDEPTH
```

2. Répétez la vérification jusqu'à ce que la longueur de la file d'attente atteigne zéro et reste à zéro lors de la vérification suivante.

La commande **REFRESH CLUSTER** supprime et recrée des objets, et dans les configurations de grande taille, l'exécution peut prendre beaucoup de temps. Voir les [remarques sur REFRESH CLUSTER pour les clusters de publication/abonnement](#).

- [REFRESH QMGR TYPE \(PROXYSUB\)](#)

Pour vérifier que la commande **REFRESH QMGR TYPE (PROXYSUB)** est terminée, vérifiez que les abonnements de proxy ont été corrigés sur d'autres gestionnaires de files d'attente du réseau distribué. Si vous utilisez le *routage direct* dans un cluster, vérifiez que les abonnements de proxy ont été corrigés sur les autres référentiels partiels du cluster. Si vous utilisez le *routage d'hôte de rubrique* dans un cluster, vérifiez que les abonnements de proxy attendus ont été corrigés sur les hôtes de rubrique correspondants. Utilisez la commande MQSC suivante:

```
DISPLAY SUB(*) SUBTYPE(PROXYSUB)
```

- [Réinitialisation d'un cluster](#)

Pour vérifier que la commande **RESET CLUSTER** est terminée, utilisez `DISPLAY CLUSQMGR`.

- [RESET QMGR TYPE \(PUBSUB\)](#)

Pour vérifier que la commande **RESET QMGR** est terminée, utilisez `DISPLAY PUBSUB TYPE (PARENT | CHILD)`.

Remarque : La commande **RESET QMGR** peut entraîner l'annulation d'abonnements de proxy dans une hiérarchie de publication / abonnement ou dans un cluster de publication / abonnement. Utilisez la commande `DISPLAY SUB SUBTYPE (PROXYSUB)` pour la surveiller.

- Vous pouvez également surveiller d'autres files d'attente système qui, au fur et à mesure de l'exécution des commandes, tendent vers une longueur de file d'attente de zéro.

Par exemple, vous pouvez surveiller la file d'attente `SYSTEM.INTER.QMGR.CONTROL` et la file d'attente `SYSTEM.INTER.QMGR.FANREQ`. Voir [Surveillance du trafic des abonnements de proxy dans les clusters et Equilibrage des producteurs et des consommateurs dans les réseaux de publication / abonnement](#).

Que faire ensuite

Si ces vérifications ne confirment pas qu'une commande asynchrone est terminée, il se peut qu'une erreur se soit produite. Pour ce faire, consultez d'abord le journal du gestionnaire de files d'attente sur lequel la commande a été émise, puis (pour un cluster) les journaux du référentiel complet du cluster.

Information associée

 z/OS

[Comportement asynchrone des commandes CLUSTER sous z/OS](#)

Utilisation de définitions de file d'attente éloignée comme alias

En plus de localiser une file d'attente sur un autre gestionnaire de files d'attente, vous pouvez également utiliser une définition locale d'une file d'attente éloignée pour les alias de gestionnaire de files d'attente et les alias de file d'attente de réponse. Les deux types d'alias sont résolus via la définition locale d'une file d'attente éloignée. Vous devez configurer les canaux appropriés pour que le message arrive à sa destination.

Alias de gestionnaire de files d'attente

Un alias est le processus par lequel le nom du gestionnaire de files d'attente cible, tel que spécifié dans un message, est modifié par un gestionnaire de files d'attente sur la route des messages. Les alias de

gestionnaire de files d'attente sont importants car vous pouvez les utiliser pour contrôler la destination des messages au sein d'un réseau de gestionnaires de files d'attente.

Pour ce faire, modifiez la définition de file d'attente éloignée sur le gestionnaire de files d'attente au point de contrôle. L'application émettrice n'est pas consciente que le nom de gestionnaire de files d'attente spécifié est un alias.

Pour plus d'informations sur les alias de gestionnaire de files d'attente, voir [Quels sont les alias?](#).

Alias de file d'attente de réponse

En option, une application peut spécifier le nom d'une file d'attente de réponse lorsqu'elle place un *message de demande* dans une file d'attente.

Si l'application qui traite le message extrait le nom de la file d'attente de réponse, elle sait où envoyer le *message de réponse*, si nécessaire.

Un alias de file d'attente de réponse est le processus par lequel une file d'attente de réponse, telle que spécifiée dans un message de demande, est modifiée par un gestionnaire de files d'attente sur la route des messages. L'application émettrice ne sait pas que le nom de file d'attente de réponse indiqué est un alias.

Un alias de file d'attente de réponse vous permet de modifier le nom de la file d'attente de réponse et éventuellement de son gestionnaire de files d'attente. Cela vous permet de contrôler la route utilisée pour les messages de réponse.

Pour plus d'informations sur les messages de demande, les messages de réponse et les files d'attente de réponse, voir [Types de message](#) et [File d'attente de réponse et gestionnaire de files d'attente](#).

Pour plus d'informations sur les alias de file d'attente de réponse, voir [Alias de file d'attente de réponse et clusters](#).

Conversion de données

Les données de message dans des formats définis par IBM MQ (également appelés formats intégrés) peuvent être converties par le gestionnaire de files d'attente d'un jeu de caractères codés à un autre, à condition que les deux jeux de caractères soient associés à une langue unique ou à un groupe de langues similaires.


Par exemple, la conversion entre les jeux de caractères codés avec les identificateurs (CCSID) 850 et 500 est prise en charge, car les deux s'appliquent aux langues d'Europe occidentale.

Pour les conversions de caractères de nouvelle ligne (NL) EBCDIC en ASCII, voir [Tous les gestionnaires de files d'attente](#).

Les conversions prises en charge sont définies dans [Traitement de la conversion de données](#).

Lorsqu'un gestionnaire de files d'attente ne parvient pas à convertir des messages dans des formats intégrés

Le gestionnaire de files d'attente ne peut pas convertir automatiquement les messages dans des formats intégrés si leurs CCSID représentent des groupes de langues nationales différents. Par exemple, la conversion entre le CCSID 850 et le CCSID 1025 (qui est un jeu de caractères codés EBCDIC pour les langues utilisant le script cyrillique) n'est pas prise en charge car la plupart des caractères d'un jeu de caractères codés ne peuvent pas être représentés dans l'autre. Si vous disposez d'un réseau de gestionnaires de files d'attente travaillant dans des langues nationales différentes et que la conversion de données entre certains jeux de caractères codés n'est pas prise en charge, vous pouvez activer une conversion par défaut.

 Pour les plateformes auxquelles le `ccsid_part2.tbl` s'applique, voir [«Spécification de la conversion de données par défaut»](#), à la page 217 using `ccsid_part2.tbl` pour plus d'informations. La conversion de données par défaut sur les plateformes autres que celles auxquelles s'applique le fichier `ccsid_part2.tbl` est décrite dans [«Conversion de données par défaut»](#), à la page 215.

Amélioration de la prise en charge de la conversion de données Unicode dans IBM MQ 9.0

V 9.0.0

Avant IBM MQ 9.0, les versions précédentes du produit ne prenaient pas en charge la conversion des données contenant des points de code Unicode au-delà du plan multilingue de base (points de code au-dessus de U+FFFF). La prise en charge de la conversion de données Unicode était limitée aux points de code définis dans la norme Unicode 3.0, codés en UTF-8 ou UCS-2, un sous-ensemble à largeur fixe de 2 octets de UTF-16.

Depuis la IBM MQ 9.0, IBM MQ prend en charge tous les caractères Unicode définis dans la norme Unicode 8.0 dans la conversion de données. Cela inclut la prise en charge complète de UTF-16, y compris les paires de substitution (une paire de caractères UTF-16 de 2 octets dans la plage X' D800 ' à X' DFFF ' qui représentent un point de code Unicode au-dessus de U+FFFF).

La combinaison de séquences de caractères est également prise en charge dans les cas où un caractère précomposé dans un CCSID est mappé à une séquence de caractères de combinaison dans un autre CCSID.

La conversion de données vers et depuis Unicode et les CCSID 1388, 1390, 1399, 4933, 5488 et 16884 a été étendue, sur certaines plateformes, pour prendre en charge tous les points de code actuellement définis pour ces CCSID, y compris ceux qui correspondent à des points de code dans des plans supplémentaires Unicode.

Dans le cas des CCSID 1390, 1399 et 16884, cela inclut les caractères définis dans la norme JIS X 0213 (JIS2004).

La prise en charge a également été ajoutée pour la conversion vers et depuis Unicode et six nouveaux CCSID (1374 à 1379).

Fichier `ccsid_part2.tbl`

V 9.0.0

A partir de IBM MQ 9.0, un fichier supplémentaire, `ccsid_part2.tbl`, est fourni.

Le fichier `ccsid_part2.tbl` a priorité sur le fichier `ccsid.tbl` et vous permet :

- D'ajouter ou de modifier des entrées de CCSID
- De spécifier une conversion de données par défaut
- De spécifier des données pour différents niveaux de commande

Le fichier `ccsid_part2.tbl` concerne uniquement les plateformes suivantes :

- **Linux** Linux - toutes les versions
- **Solaris** Solaris
- **Windows** Windows

Windows Depuis IBM MQ 9.0, sous IBM MQ for Windows, `ccsid_part2.tbl` se trouve dans le répertoire `MQDataRoot\conv\table` par défaut. En outre, sous IBM MQ for Windows, il enregistre tous les jeux de codes pris en charge.

Solaris **Linux** Depuis IBM MQ 9.0, sur les plateformes IBM MQ for Linux et Solaris, `ccsid_part2.tbl` se trouve dans le répertoire `MQDataRoot/conv/table`. Pour toutes les plateformes Linux et Solaris, les jeux de codes pris en charge sont conservés dans les tables de conversion fournies par IBM MQ.

Bien que le fichier `ccsid_part2.tbl` remplace le fichier `ccsid.tbl` existant utilisé dans les versions précédentes de IBM MQ pour fournir des informations CCSID supplémentaires, le fichier `ccsid.tbl` continue d'être analysé par IBM MQ et ne doit donc pas être supprimé.

Pour plus d'informations, voir «[Le fichier ccsid_part2.tbl](#)», à la page 216.

fichier ccsid.tbl

V 9.0.0 Sur les plateformes autres que celles auxquelles `ccsid_part2.tbl` s'applique, le fichier `ccsid.tbl` est utilisé aux fins suivantes:

- **HP-UX** **AIX** Sur les plateformes AIX et HP-UX, les jeux de codes pris en charge sont conservés en interne par le système d'exploitation.
- Il indique tous les jeux de codes supplémentaires. Pour spécifier des jeux de codes supplémentaires, vous devez éditer `ccsid.tbl` (des conseils sur la procédure à suivre sont fournis dans le fichier).
- Il indique toute conversion de données par défaut.

Vous pouvez mettre à jour les informations enregistrées dans `ccsid.tbl`. Vous pouvez être amené à le faire si, par exemple, une version ultérieure de votre système d'exploitation prend en charge des jeux de caractères codés supplémentaires.

Conversion de données par défaut

V 9.0.0 Depuis la IBM MQ 9.0, la méthode de conversion des données par défaut a été modifiée sur les plateformes suivantes:

- Linux - toutes les versions
- Solaris
- Windows

Pour plus d'informations, voir «[Spécification de la conversion de données par défaut](#)», à la page 217 Utilisation de `ccsid_part2.tbl`.

Si vous configurez des canaux entre deux machines sur lesquelles la conversion de données n'est pas normalement prise en charge, vous devez activer la conversion de données par défaut pour que les canaux fonctionnent.

V 9.0.0 Sur les plateformes autres que celles auxquelles `ccsid_part2.tbl` s'applique, pour activer la conversion de données par défaut, éditez le fichier `ccsid.tbl` pour spécifier un CCSID EBCDIC par défaut et un CCSID ASCII par défaut. Des instructions sur la façon de procéder sont incluses dans le fichier. Vous devez effectuer cette opération sur toutes les machines qui seront connectées à l'aide des canaux. Redémarrez le gestionnaire de files d'attente pour que la modification soit prise en compte.

Le processus de conversion de données par défaut est le suivant:

- Si la conversion entre les CCSID source et cible n'est pas prise en charge, mais que les CCSID des environnements source et cible sont EBCDIC ou ASCII, les données de type caractères sont transmises à l'application cible sans conversion.
- Si un CCSID représente un jeu de caractères codés ASCII et que l'autre représente un jeu de caractères codés EBCDIC, IBM MQ convertit les données à l'aide des CCSID de conversion de données par défaut définis dans `ccsid.tbl`.

Remarque : Essayez de limiter les caractères convertis à ceux qui ont les mêmes valeurs de code dans le jeu de caractères codés indiqué pour le message et dans le jeu de caractères codés par défaut. Si vous utilisez uniquement l'ensemble de caractères qui est valide pour les noms d'objet IBM MQ (tel que défini dans [Nommage des objets IBM MQ](#)) vous satisferez, en général, à cette exigence. Des exceptions se produisent avec les CCSID EBCDIC 290, 930, 1279 et 5026 utilisés au Japon, où les caractères minuscules ont des codes différents de ceux utilisés dans les autres CCSID EBCDIC.

Conversion de messages dans des formats définis par l'utilisateur

Le gestionnaire de files d'attente ne peut pas convertir des messages dans des formats définis par l'utilisateur d'un jeu de caractères codés à un autre. Si vous devez convertir des données dans un

format défini par l'utilisateur, vous devez fournir un exit de conversion de données pour chaque format. N'utilisez pas les CCSID par défaut pour convertir les données de type caractères dans des formats définis par l'utilisateur. Pour plus d'informations sur la conversion de données dans des formats définis par l'utilisateur et sur l'écriture d'exits de conversion de données, voir [Ecriture d'exits de conversion de données](#).

Modification du CCSID du gestionnaire de files d'attente

Une fois que vous avez utilisé l'attribut CCSID de la commande ALTER QMGR pour modifier le CCSID du gestionnaire de files d'attente, arrêtez et redémarrez le gestionnaire de files d'attente pour vous assurer que toutes les applications en cours d'exécution, y compris le serveur de commandes et les programmes de canal, sont arrêtées et redémarrées.

Cette opération est nécessaire car toutes les applications en cours d'exécution lorsque le CCSID du gestionnaire de files d'attente est modifié continuent d'utiliser le CCSID existant.

Le fichier `ccsid_part2.tbl`

Depuis IBM MQ 9.0, le fichier `ccsid_part2.tbl` remplace le fichier `ccsid.tbl` existant utilisé dans les versions précédentes du produit, afin de fournir des informations CCSID supplémentaires. Notez que le fichier `ccsid.tbl` continue d'être analysé par IBM MQ et ne doit pas être supprimé, et que les entrées de `ccsid_part2.tbl` ont priorité sur les autres entrées de `ccsid.tbl`.

Le fichier `ccsid_part2.tbl` se trouve dans le répertoire `MQDataRoot/conv/table` sur:

- Linux - toutes les versions
- Solaris

et le répertoire `MQDataRoot\conv\table` sous Windows

Notez que le fichier `ccsid_part2.tbl` n'est disponible sur aucune autre plateforme. Le fichier est utilisé pour effectuer les actions suivantes:

- Ajout ou modification d'entrées CCSID existantes à utiliser dans la conversion de données IBM MQ
- Spécification de la conversion de données par défaut

Vous devez utiliser `ccsid_part2.tbl` comme fichier suivant:

- Contient la prise en charge des nouvelles valeurs de codage Unicode.
- Permet d'indiquer la version des entrées CCSID de sorte qu'elles ne s'appliquent qu'aux niveaux de commande sélectionnés.

Ajout ou modification d'entrées CCSID

Une entrée du fichier `ccsid_part2.tbl` a le format suivant:

```
<CCSID number> <Base CCSID> <DBCS CodePage> <SBCS CodePage>  
<Type> <Encoding> <ACRI> <Name>
```

Voici un exemple d'entrée pour CCSID 1200 (UTF-16):

```
1200 1200 1200 1200 3 8 0 UTF-16
```

Remarque : Pour plus de détails sur la valeur d'ACRI, voir le commentaire dans le fichier `ccsid_part2.tbl`.

Au format `ccsid_part2.tbl` :

Le type peut être égal à:

- 1=SBCS
- 2=DBCS

3=MBCS

Le codage peut être égal à:

1=EBCDIC

2 = ASCII

3 = ISO

4 = UCS-2

5 = UTF-8

6 = Euc

7 = GB18030

8 = UTF-16

9 = UTF-32

Lorsque vous éditez le fichier, vous:

- Peut spécifier un commentaire en utilisant le symbole # au début d'une ligne. Cela empêche IBM MQ de tenter d'analyser la ligne.
- Impossible de fournir des commentaires en ligne.
- Vous devez vous assurer que vous ne créez pas de lignes vides.
- Vous ne devez pas ajouter de nouvelles entrées à la fin du fichier.

De nouvelles entrées CCSID doivent être ajoutées avant les informations de la table ACRI.

Spécification de la conversion de données par défaut

Vous pouvez définir des CCSID de conversion par défaut, qui sont utilisés pour la conversion entre des CCSID ASCII ou similaires et des CCSID EBCDIC, si aucune conversion n'est prise en charge entre deux CCSID.

Si vous activez cette fonction, la conversion par défaut est utilisée pour la transmission et les en-têtes de message, et peut également être utilisée dans la conversion des données utilisateur.

Les conversions par défaut sont activées en créant deux lignes similaires aux suivantes:

```
default      0      500      1      1      0
default      0      850      1      2      0
```

La première ligne définit la valeur par défaut pour les CCSID EBCDIC sur 500 et la deuxième ligne définit la valeur par défaut pour les CCSID ASCII et similaires sur 850.

Spécification de données pour différents niveaux de commande

Pour spécifier des entrées CCSID pour les différents niveaux de commande de IBM MQ , utilisez un symbole deux points suivi du niveau de commande (ou des niveaux de commande) de IBM MQ auquel vous souhaitez que la section suivante soit applicable.

Ce nombre représente le niveau de commande minimal dans lequel le gestionnaire de files d'attente ou le client doit s'exécuter. Par exemple, si le gestionnaire de files d'attente en cours est au niveau de commande 900 et qu'il rencontre un indicateur de niveau de commande 800 ou 900, les CCSID sont lus.

Toutefois, un gestionnaire de files d'attente de niveau 800 ignore les CCSID de la section 900.

Le niveau de commande indiqué est applicable à toutes les entrées CCSID détectées après un indicateur de niveau de commande, jusqu'à ce qu'un nouvel indicateur de niveau de commande soit trouvé.

Si vous devez définir le niveau de commande sur tous les niveaux de commande, indiquez le nombre zéro.

Lors de la première analyse syntaxique de `ccsid_part2.tbl`, IBM MQ traite tous les CCSID détectés comme valides pour tous les niveaux de commande de IBM MQ.

La gestion des versions commence à être utilisée uniquement lorsque IBM MQ rencontre le premier indicateur de niveau de commande.

Le fragment de code suivant illustre un exemple d'utilisation de la gestion des versions:

```
# Comment Block
# End of Comment Block
# Because no command level flag is specified and we're at the start of the file
# the following CCSIDs will be read on all versions
  819  819    0    819    1    3    0    IS08859-1
  923  923    0    923    1    3    0    IS08859-15
 1051 1051    0    1051   1    3    0    IBM-1051
# The colon :900 below shows that the CCSIDs after will only be for MQ cmd level 900 and above
:900
  8629 437    0    437    1    2    0    IBM-437
 12725 437    0    437    1    2    0    IBM-437
 16821 437    0    437    1    2    0    IBM-437
 20917 437    0    437    1    2    0    IBM-437
# The colon :0 below shows that the CCSIDs after will be for all version of MQ
:0
  4946 850    0    850    1    2    0    IBM-850
 33618 850    0    850    1    2    0    IBM-850
 61697 850    0    850    1    2    0    IBM-850
 61698 850    0    850    1    2    0    IBM-850
```

Windows

Linux

AIX

Administration d'MQ Telemetry

MQ Telemetry est administré à l'aide de IBM MQ Explorer ou d'une ligne de commande. Utilisez l'explorateur pour configurer les canaux de télémétrie, contrôler le service de télémétrie et surveiller les clients MQTT qui sont connectés à IBM MQ. Configurez la sécurité de MQ Telemetry à l'aide de JAAS, de TLS et du gestionnaire des droits d'accès aux objets IBM MQ .

Administration à l'aide de IBM MQ Explorer

Utilisez l'explorateur pour configurer les canaux de télémétrie, contrôler le service de télémétrie et surveiller les clients MQTT qui sont connectés à IBM MQ. Configurez la sécurité de MQ Telemetry à l'aide de JAAS, de TLS et du gestionnaire des droits d'accès aux objets IBM MQ .

Administration à l'aide de la ligne de commande

MQ Telemetry peut être entièrement administré à partir de la ligne de commande à l'aide des commandes IBM MQ [MQSC](#) .

La documentation MQ Telemetry contient également des exemples de script qui illustrent l'utilisation de base de l'application client IBM MQ Telemetry Transport v3 .

Lisez et comprenez les exemples dans les [exemples de programmes IBM MQ Telemetry Transport](#) avant de les utiliser.

Information associée

[MQ Telemetry](#)

[Propriétés de MQXR](#)

Linux

AIX

Configuration d'un gestionnaire de files d'attente pour la télémétrie sous Linux et AIX

Suivez ces étapes manuelles pour configurer un gestionnaire de files d'attente afin qu'il exécute MQ Telemetry. Vous pouvez exécuter une procédure automatisée pour définir une configuration plus simple à l'aide du support MQ Telemetry pour IBM MQ Explorer.

Avant de commencer

1. Pour plus d'informations sur l'installation de IBM MQ et de la fonction MQ Telemetry , voir [Installation de MQ Telemetry](#) .

2. Créez et démarrez un gestionnaire de files d'attente. Le gestionnaire de files d'attente est appelé *qMgr* dans cette tâche.
3. Dans le cadre de cette tâche, vous configurez le service de télémétrie (MQXR). Les paramètres de propriété MQXR sont stockés dans un fichier de propriétés spécifique à la plateforme: `mqxr_unix.properties`. En règle générale, vous n'avez pas besoin d'éditer directement le fichier de propriétés MQXR, car la plupart des paramètres peuvent être configurés via des commandes d'administration MQSC ou IBM MQ Explorer. Si vous décidez d'éditer le fichier directement, arrêtez le gestionnaire de files d'attente avant d'effectuer vos modifications. Voir [MQXR properties](#).

Pourquoi et quand exécuter cette tâche

La MQ Telemetry prise en charge de IBM MQ Explorer inclut un assistant et un exemple de procédure de commande `sampleMQM`. Ils définissent une configuration initiale à l'aide de l'ID utilisateur invité ; voir [Vérification de l'installation de MQ Telemetry à l'aide des exemples de programme IBM MQ Explorer et IBM MQ Telemetry Transport](#).

Suivez les étapes de cette tâche pour configurer MQ Telemetry manuellement à l'aide de différents schémas d'autorisation.

Procédure

1. Ouvrez une fenêtre de commande dans le répertoire des exemples de télémétrie.
Le répertoire des exemples de télémétrie est `/opt/mqm/mqxr/samples`.
2. Créez la file d'attente de transmission de télémétrie.

```
echo "DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000)" | runmqsc qMgr
```

Lorsque le service de télémétrie (MQXR) est démarré pour la première fois, il crée `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Il est créé manuellement dans cette tâche, car `SYSTEM.MQTT.TRANSMIT.QUEUE` doit exister avant le démarrage du service de télémétrie (MQXR) pour autoriser l'accès à ce service.

3. Définir la file d'attente de transmission par défaut

Lorsque le service de télémétrie (MQXR) est démarré pour la première fois, il ne modifie pas le gestionnaire de files d'attente pour faire de `SYSTEM.MQTT.TRANSMIT.QUEUE` la file d'attente de transmission par défaut.

Pour que `SYSTEM.MQTT.TRANSMIT.QUEUE` soit la file d'attente de transmission par défaut, modifiez la propriété de file d'attente de transmission par défaut. Modifiez la propriété à l'aide de IBM MQ Explorer ou de la commande de l'exemple suivant:

```
echo "ALTER QMGR DEFQXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE')" | runmqsc qMgr
```

La modification de la file d'attente de transmission par défaut peut interférer avec votre configuration existante. La modification de la file d'attente de transmission par défaut en `SYSTEM.MQTT.TRANSMIT.QUEUE` a pour but de faciliter l'envoi de messages directement aux clients MQTT. Sans modifier la file d'attente de transmission par défaut, vous devez ajouter une définition de file d'attente éloignée pour chaque client qui reçoit des messages IBM MQ Explorer ; voir [«Envoyer direct d'un message à un client»](#), à la page 224.

4. Suivez une procédure dans [«Autorisation des clients MQTT à accéder aux objets IBM MQ»](#), à la page 226 pour créer un ou plusieurs ID utilisateur. Les ID utilisateur ont le droit de publier, de s'abonner et d'envoyer des publications aux clients MQTT.
5. Installation du service de télémétrie (MQXR)

```
cat /opt/<install_dir>/mqxr/samples/installMQXRService_unix.mqsc | runmqsc qMgr
```

Voir aussi l'exemple de code dans [Figure 20](#), à la page 220.

6. Démarrez le service

```
echo "START SERVICE(SYSTEM.MQXR.SERVICE)" | runmqsc qMgr
```

Le service de télémétrie (MQXR) est démarré automatiquement au démarrage du gestionnaire de files d'attente.

Il est démarré manuellement dans cette tâche, car le gestionnaire de files d'attente est déjà en cours d'exécution.

7. A l'aide de IBM MQ Explorer, configurez les canaux de télémétrie pour accepter les connexions des clients MQTT .

Les canaux de télémétrie doivent être configurés de sorte que leurs identités soient l'un des ID utilisateur définis à l'étape 4.

Voir aussi [DEFINE CHANNEL \(MQTT\)](#).

8. Vérifiez la configuration en exécutant l'exemple de client.

Pour que l'exemple de client puisse utiliser votre canal de télémétrie, le canal doit autoriser le client à publier, s'abonner et recevoir des publications. L'exemple de client se connecte au canal de télémétrie sur le port 1883 par défaut. Voir aussi [exemples de programmes IBM MQ Telemetry Transport](#).

Exemple

La [Figure 20](#), à la page 220 illustre la commande `runmqsc` permettant de créer le `SYSTEM.MQXR.SERVICE` manuellement sous Linux.

```
DEF SERVICE(SYSTEM.MQXR.SERVICE) +
CONTROL(QMGR) +
DESCR('Manages clients using MQXR protocols such as MQTT') +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+/mqxr/bin/runMQXRService.sh') +
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+" -g "+MQ_DATA_PATH+"') +
STOPCMD('+MQ_INSTALL_PATH+/mqxr/bin/endMQXRService.sh') +
STOPARG('-m +QMNAME+') +
STDOUT('+MQ_Q_MGR_DATA_PATH+/mqxr.stdout') +
STDERR('+MQ_Q_MGR_DATA_PATH+/mqxr.stderr')
```

Figure 20. `installMQXRService_unix.mqsc`

Windows Configuration d'un gestionnaire de files d'attente pour la télémétrie sous Windows

Suivez ces étapes manuelles pour configurer un gestionnaire de files d'attente afin qu'il exécute MQ Telemetry. Vous pouvez exécuter une procédure automatisée pour définir une configuration plus simple à l'aide du support MQ Telemetry pour IBM MQ Explorer.

Avant de commencer

1. Pour plus d'informations sur l'installation de IBM MQ et de la fonction MQ Telemetry, voir [Installation de MQ Telemetry](#).
2. Créez et démarrez un gestionnaire de files d'attente. Le gestionnaire de files d'attente est appelé `qMgr` dans cette tâche.
3. Dans le cadre de cette tâche, vous configurez le service de télémétrie (MQXR). Les paramètres de propriété MQXR sont stockés dans un fichier de propriétés spécifique à la plateforme: `mqxr_win.properties`. En règle générale, vous n'avez pas besoin d'éditer directement le fichier de propriétés MQXR, car la plupart des paramètres peuvent être configurés via des commandes d'administration MQSC ou IBM MQ Explorer. Si vous décidez d'éditer le fichier directement, arrêtez le gestionnaire de files d'attente avant d'effectuer vos modifications. Voir [MQXR properties](#).

Pourquoi et quand exécuter cette tâche

La MQ Telemetry prise en charge de IBM MQ Explorer inclut un assistant et un exemple de procédure de commande `sampleMQM`. Ils définissent une configuration initiale à l'aide de l'ID utilisateur invité ; voir [Vérification de l'installation de MQ Telemetry à l'aide des exemples de programme IBM MQ Explorer et IBM MQ Telemetry Transport](#).

Suivez les étapes de cette tâche pour configurer MQ Telemetry manuellement à l'aide de différents schémas d'autorisation.

Procédure

1. Ouvrez une fenêtre de commande dans le répertoire des exemples de télémétrie.

Le répertoire des exemples de télémétrie est `WMQ program installation directory\mqxr\samples`.

2. Créez la file d'attente de transmission de télémétrie.

```
echo DEFINE QLOCAL('SYSTEM.MQTT.TRANSMIT.QUEUE') USAGE(XMITQ) MAXDEPTH(100000) | runmqsc qMgr
```

Lorsque le service de télémétrie (MQXR) est démarré pour la première fois, il crée `SYSTEM.MQTT.TRANSMIT.QUEUE`.

Il est créé manuellement dans cette tâche, car `SYSTEM.MQTT.TRANSMIT.QUEUE` doit exister avant le démarrage du service de télémétrie (MQXR) pour autoriser l'accès à ce service.

3. Définition de la file d'attente de transmission par défaut pour `qMgr`

```
echo ALTER QMGR DEFXMITQ('SYSTEM.MQTT.TRANSMIT.QUEUE') | runmqsc qMgr
```

Figure 21. Définir la file d'attente de transmission par défaut

Lorsque le service de télémétrie (MQXR) est démarré pour la première fois, il ne modifie pas le gestionnaire de files d'attente pour faire de `SYSTEM.MQTT.TRANSMIT.QUEUE` la file d'attente de transmission par défaut.

Pour que `SYSTEM.MQTT.TRANSMIT.QUEUE` soit la file d'attente de transmission par défaut, modifiez la propriété de file d'attente de transmission par défaut. Modifiez la propriété à l'aide de IBM MQ Explorer ou de la commande dans [Figure 21](#), à la page 221.

La modification de la file d'attente de transmission par défaut peut interférer avec votre configuration existante. La modification de la file d'attente de transmission par défaut en `SYSTEM.MQTT.TRANSMIT.QUEUE` a pour but de faciliter l'envoi de messages directement aux clients MQTT. Sans modifier la file d'attente de transmission par défaut, vous devez ajouter une définition de file d'attente éloignée pour chaque client qui reçoit des messages IBM MQ ; voir [«Envoi direct d'un message à un client»](#), à la page 224.

4. Suivez une procédure dans [«Autorisation des clients MQTT à accéder aux objets IBM MQ»](#), à la page 226 pour créer un ou plusieurs ID utilisateur. Les ID utilisateur ont le droit de publier, de s'abonner et d'envoyer des publications aux clients MQTT.
5. Installation du service de télémétrie (MQXR)

```
type  
installMQXRService_win.mqsc | runmqsc qMgr
```

6. Démarrez le service

```
echo START SERVICE(SYSTEM.MQXR.SERVICE) | runmqsc qMgr
```

Le service de télémétrie (MQXR) est démarré automatiquement au démarrage du gestionnaire de files d'attente.

Il est démarré manuellement dans cette tâche, car le gestionnaire de files d'attente est déjà en cours d'exécution.

7. A l'aide de IBM MQ Explorer, configurez les canaux de télémétrie pour accepter les connexions des clients MQTT .

Les canaux de télémétrie doivent être configurés de sorte que leurs identités soient l'un des ID utilisateur définis à l'étape 4.

Voir aussi [DEFINE CHANNEL \(MQTT\)](#).

8. Vérifiez la configuration en exécutant l'exemple de client.

Pour que l'exemple de client puisse utiliser votre canal de télémétrie, le canal doit autoriser le client à publier, s'abonner et recevoir des publications. L'exemple de client se connecte au canal de télémétrie sur le port 1883 par défaut. Voir aussi [exemples de programmes IBM MQ Telemetry Transport](#).

Création manuelle de SYSTEM.MQXR.SERVICE

La Figure 22, à la page 222 illustre la commande `runmqsc` permettant de créer le SYSTEM.MQXR.SERVICE manuellement sous Windows.

```
DEF SERVICE(SYSTEM.MQXR.SERVICE) +
CONTROL(QMGR) +
DESCR('Manages clients using MQXR protocols such as MQTT') +
SERVTYPE(SERVER) +
STARTCMD('+MQ_INSTALL_PATH+mqxr\bin\runMQXRService.bat') +
STARTARG('-m +QMNAME+ -d "+MQ_Q_MGR_DATA_PATH+\" -g "+MQ_DATA_PATH+\"') +
STOPCMD('+MQ_INSTALL_PATH+mqxr\bin\endMQXRService.bat') +
STOPARG('-m +QMNAME+') +
STDOUT('+MQ_Q_MGR_DATA_PATH+mqxr.stdout') +
STDERR('+MQ_Q_MGR_DATA_PATH+mqxr.stderr')
```

Figure 22. `installMQXRService_win.mqsc`

Windows Linux AIX Configuration de la mise en file d'attente répartie pour l'envoi de messages aux clients MQTT

Les applications IBM MQ peuvent envoyer des messages aux clients MQTT v3 en les publiant dans un abonnement créé par un client ou en envoyant un message directement. Quelle que soit la méthode utilisée, le message est placé sur SYSTEM.MQTT.TRANSMIT.QUEUE et envoyé au client par le service de télémétrie (MQXR). Il existe plusieurs façons de placer un message sur SYSTEM.MQTT.TRANSMIT.QUEUE.

Publication d'un message en réponse à un abonnement client MQTT

Le service de télémétrie (MQXR) crée un abonnement pour le compte du client MQTT . Le client est la destination des publications qui correspondent à l'abonnement envoyé par le client. Les services de télémétrie réachemine les publications correspondantes vers le client.

Un client MQTT est connecté à IBM MQ en tant que gestionnaire de files d'attente, son nom de gestionnaire de files d'attente étant défini sur `ClientIdentifier`. La destination des publications à envoyer au client est une file d'attente de transmission, SYSTEM.MQTT.TRANSMIT.QUEUE. Le service de télémétrie transmet les messages sur SYSTEM.MQTT.TRANSMIT.QUEUE aux clients MQTT , en utilisant le nom du gestionnaire de files d'attente cible comme clé d'un client spécifique.

Le service de télémétrie (MQXR) ouvre la file d'attente de transmission en utilisant `ClientIdentifier` comme nom de gestionnaire de files d'attente. Le service de télémétrie (MQXR) transmet le descripteur d'objet de la file d'attente à l'appel `MQSUB` pour transmettre les publications qui correspondent à l'abonnement du client. Dans la résolution de nom d'objet, `ClientIdentifier` est créé en tant que nom de gestionnaire de files d'attente éloignées et la file d'attente de transmission doit être résolue en SYSTEM.MQTT.TRANSMIT.QUEUE. A l'aide de la résolution de nom d'objet IBM MQ standard, `ClientIdentifier` est résolu comme suit ; voir [Tableau 14, à la page 223](#).

1. *ClientIdentifier* ne correspond à rien.

ClientIdentifier est un nom de gestionnaire de files d'attente éloignées. Il ne correspond pas au nom du gestionnaire de files d'attente local, à un alias de gestionnaire de files d'attente ou à un nom de file d'attente de transmission.

Le nom de la file d'attente n'est pas défini. Actuellement, le service de télémétrie (MQXR) définit SYSTEM.MQTT.PUBLICATION.QUEUE comme nom de la file d'attente. Un client MQTT v3 ne prenant pas en charge les files d'attente, le nom de la file d'attente résolue est ignoré par le client. La propriété du gestionnaire de files d'attente local, File d'attente de transmission par défaut, doit être définie sur SYSTEM.MQTT.TRANSMIT.QUEUE, de sorte que la publication soit placée sur SYSTEM.MQTT.TRANSMIT.QUEUE pour être envoyée au client.

2. *ClientIdentifier* correspond à un alias de gestionnaire de files d'attente nommé *ClientIdentifier*.

ClientIdentifier est un nom de gestionnaire de files d'attente éloignées. Il correspond au nom d'un alias de gestionnaire de files d'attente.

L'alias de gestionnaire de files d'attente doit être défini avec *ClientIdentifier* comme nom de gestionnaire de files d'attente éloignées.

En définissant le nom de la file d'attente de transmission dans la définition d'alias de gestionnaire de files d'attente, il n'est pas nécessaire que la transmission par défaut soit définie sur SYSTEM.MQTT.TRANSMIT.QUEUE.

| | Entrée | | Sortie | | |
|--|-------------------------|--------------------------|-------------------------|--------------------------|--|
| | Nom gest. de files | Nom de la file d'attente | Nom gest. de files | Nom de la file d'attente | File d'attente de transmission |
| Ne correspond à rien | <i>ClientIdentifier</i> | <i>non défini</i> | <i>ClientIdentifier</i> | <i>non défini</i> | File d'attente de transmission par défaut. SYSTEM.MQTT.TRANSMIT.QUEUE |
| Correspond à un alias de gestionnaire de files d'attente nommé <i>ClientIdentifier</i> | <i>ClientIdentifier</i> | <i>non défini</i> | <i>ClientIdentifier</i> | <i>non défini</i> | SYSTEM.MQTT.TRANSMIT.QUEUE |

Pour plus d'informations sur la résolution de nom, voir [Résolution de nom](#).

Tout programme IBM MQ peut publier dans la même rubrique. La publication est envoyée à ses abonnés, y compris les clients MQTT v3 qui ont un abonnement à la rubrique.

Si une rubrique d'administration est créée dans un cluster, avec l'attribut CLUSTER(*clusterName*), toute application du cluster peut être publiée sur le client ; par exemple:

```
echo DEFINE TOPIC('MQTTExamples') TOPICSTR('MQTT Examples') CLUSTER(MQTT) REPLACE | runmqsc qMgr
```

Figure 23. Définition d'une rubrique de cluster sous Windows

Remarque : N'attribuez pas à SYSTEM.MQTT.TRANSMIT.QUEUE un attribut de cluster.

Les abonnés du client MQTT et les diffuseurs de publications peuvent se connecter à différents gestionnaires de files d'attente. Les abonnés et les diffuseurs peuvent faire partie du même cluster ou être connectés par une hiérarchie de publication / abonnement. La publication est distribuée par le diffuseur de publications à l'abonné à l'aide de IBM MQ.

Envoi direct d'un message à un client

Alternative à un client qui crée un abonnement et qui reçoit une publication correspondant à la rubrique d'abonnement, qui envoie directement un message à un client MQTT v3. Les applications client MQTT V3 ne peuvent pas envoyer de messages directement, contrairement à d'autres applications, telles que les applications IBM MQ.

L'application IBM MQ doit connaître `ClientIdentifieur` du client MQTT v3. Comme les clients MQTT v3 n'ont pas de files d'attente, le nom de la file d'attente cible est transmis à la méthode `messageArrived` du client d'application MQTT v3 en tant que nom de rubrique. Par exemple, dans un programme MQI, créez un descripteur d'objet avec le client comme `ObjectQmgrNom`:

```
MQOD.ObjectQmgrName = ClientIdentifieur ;
MQOD.ObjectName = name ;
```

Figure 24. Descripteur d'objet MQI permettant d'envoyer un message à une destination client MQTT v3

Si l'application est écrite à l'aide de JMS, créez une destination point à point, par exemple:

```
javax.jms.Destination jmsDestination =
(javax.jms.Destination)jmsFactory.createQueue
("queue://ClientIdentifieur/name");
```

Figure 25. Destination JMS pour l'envoi d'un message à un client MQTT v3

Pour envoyer un message non sollicité à un client MQTT, utilisez une définition de file d'attente éloignée. Le nom du gestionnaire de files d'attente éloignées doit être résolu en `ClientIdentifieur` du client. La file d'attente de transmission doit être résolue en `SYSTEM.MQTT.TRANSMIT.QUEUE`; voir [Tableau 15](#), à la page 224. Le nom de la file d'attente éloignée peut être n'importe quel nom. Le client le reçoit sous forme de chaîne de rubrique.

| Entrée | | Sortie | | |
|---|--|---|--------------------------------|---|
| Nom de la file d'attente | Nom gest. de files | Nom de la file d'attente | Nom gest. de files | File d'attente de transmission |
| Nom de la définition de file d'attente éloignée | Nom de gestionnaire de files d'attente local ou vide | Nom de file d'attente éloignée utilisé comme chaîne de rubrique | <code>ClientIdentifieur</code> | <code>SYSTEM.MQTT.TRANSMIT.QUEUE</code> |

Si le client est connecté, le message est envoyé directement au client MQTT, qui appelle la méthode `messageArrived`; voir [MéthodemessageArrived](#).

Si le client s'est déconnecté avec une session persistante, le message est stocké dans `SYSTEM.MQTT.TRANSMIT.QUEUE`; voir [MQTT sessions sans état et avec état](#). Il est transmis au client lorsque ce dernier se reconnecte à nouveau à la session.

Si vous envoyez un message non persistant, il est envoyé au client avec la qualité de service au plus une fois, `QoS=0`. Si vous envoyez un message persistant directement à un client, par défaut, il est

envoyé avec la qualité de service une seule fois, QoS=2. Comme le client n'a peut-être pas de mécanisme de persistance, il peut réduire la qualité de service qu'il accepte pour les messages envoyés directement. Pour réduire la qualité de service des messages envoyés directement à un client, abonnez-vous à la rubrique DEFAULT . QoS. Indiquez la qualité de service maximale que le client peut prendre en charge.

Windows

Linux

AIX

MQTT Identification, autorisation et

authentification du client

Le service de télémétrie (MQXR) publie des rubriques IBM MQ ou s'y abonne pour le compte de clients MQTT, à l'aide de canaux MQTT. L'administrateur IBM MQ configure l'identité de canal MQTT qui est utilisée pour l'autorisation IBM MQ. L'administrateur peut définir une identité commune pour le canal ou utiliser le Nom d'utilisateur ou l'Identificateur client d'un client connecté au canal.

Le service de télémétrie (MQXR) peut authentifier le client à l'aide du Nom d'utilisateur fourni par le client ou à l'aide d'un certificat client. Le Nom d'utilisateur est authentifié à l'aide d'un mot de passe fourni par le client.

En résumé : l'identification client est la sélection de l'identité du client. En fonction du contexte, le client est identifié par l'Identificateur client, le Nom d'utilisateur, une identité client commune créée par l'administrateur ou un certificat client. L'identificateur client utilisé pour le contrôle d'authenticité ne doit pas être le même que celui utilisé pour l'autorisation.

Les programmes client MQTT définissent le nom d'utilisateur et le mot de passe qui sont envoyés au serveur à l'aide d'un canal MQTT. Ils peuvent également définir les propriétés TLS requises pour chiffrer et authentifier la connexion. L'administrateur décide si le canal MQTT doit être authentifié et comment l'authentifier.

Pour autoriser un client MQTT à accéder à des objets IBM MQ, autorisez ClientIdentifier ou Username du client ou autorisez une identité client commune. Pour autoriser un client à se connecter à IBM MQ, vous devez authentifier le nom d'utilisateur ou utiliser un certificat client. Configurez JAAS pour authentifier le nom d'utilisateur et configurez TLS pour authentifier un certificat client.

Si vous définissez un mot de passe sur le client, chiffrez la connexion à l'aide de VPN ou configurez le canal MQTT pour qu'il utilise TLS afin de conserver le mot de passe privé.

Il est difficile de gérer des certificats client. Pour cette raison, si les risques associés à l'authentification par mot de passe sont acceptables, ce type d'authentification est souvent utilisé pour l'authentification des clients.

S'il existe un moyen sûr de gérer et de stocker le certificat client, il est possible de faire confiance à l'authentification par certificat. Toutefois, il arrive rarement que les certificats puissent être gérés de manière sécurisée dans les types d'environnements dans lesquels la télémétrie est utilisée. L'authentification des dispositifs utilisant les certificats client est complétée par l'authentification des mots de passe client au niveau du serveur. En raison de la complexité supplémentaire, l'utilisation du certificat client est limitée aux applications hautement sensibles. L'utilisation de deux formulaires d'authentification est appelée authentification à deux facteurs. Vous devez connaître l'un des facteurs, tels que le mot de passe, et disposer de l'autre, un certificat, par exemple.

Dans une application sensible, telle qu'un appareil utilisant un code confidentiel, l'appareil est verrouillé pendant sa fabrication pour empêcher toute contrefaçon avec le matériel et le logiciel internes. Un certificat client digne de confiance, limité dans le temps, est copié dans le dispositif. Le dispositif est déployé à l'emplacement où il doit être utilisé. Une authentification supplémentaire est effectuée chaque fois que le dispositif est utilisé, soit à l'aide d'un mot de passe, soit en utilisant un autre certificat à partir d'une carte à puce.

Windows

Linux

AIX

MQTT Identité et autorisation du client

Utilisez l'ID client, Username, ou une identité de client commune pour l'autorisation d'accès aux objets IBM MQ.

L'administrateur IBM MQ dispose de trois options pour sélectionner l'identité du canal MQTT . L'administrateur fait le choix lors de la définition ou de la modification du canal MQTT utilisé par le client. L'identité est utilisée pour autoriser l'accès aux rubriques IBM MQ. Le choix se fait dans l'ordre suivant:

1. L'ID client (voir [USECLNTID](#)).
2. Identité fournie par l'administrateur pour le canal (MCAUSER du canal. Voir [MCAUSER](#)).
3. Si aucun des choix précédents ne s'applique, le Nom d'utilisateur transmis par le client MQTT (Nom d'utilisateur est un attribut de la classe `MqttConnectOptions` . Il doit être défini avant que le client ne se connecte au service. Sa valeur par défaut est null).

Éviter les incidents : L'identité choisie par ce processus est ensuite désignée, par exemple par la commande `DISPLAY CHSTATUS (MQTT)`, comme MCAUSER du client. Sachez qu'il ne s'agit pas nécessairement de la même identité que le MCAUSER du canal auquel il est fait référence dans le choix (2).

La commande IBM MQ `setmqaut` permet de sélectionner les objets et les actions autorisés à être utilisés par l'identité associée au canal MQTT . Par exemple, le code suivant autorise une identité de canal `MQTTClient`, fournie par l'administrateur du gestionnaire de files d'attente `QM1`:

```
setmqaut -m QM1 -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p MQTTClient -all +put
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p MQTTClient -all +pub +sub
```

Windows

Linux

AIX

Autorisation des clients MQTT à accéder aux objets

IBM MQ

Procédez comme suit pour autoriser les clients MQTT à publier des objets IBM MQ et à s'y abonner. Ces étapes décrivent quatre modèles de contrôle d'accès différents.

Avant de commencer

Les clients MQTT sont autorisés à accéder aux objets dans IBM MQ en se faisant affecter une identité lorsqu'ils se connectent à un canal de télémétrie. L'administrateur IBM MQ configure le canal de télémétrie à l'aide de IBM MQ Explorer pour donner à un client l'un des trois types d'identité suivants:

1. `ClientIdentifier`
2. Nom d'utilisateur
3. Nom que l'administrateur affecte au canal.

Quel que soit le type utilisé, l'identité doit être définie dans IBM MQ en tant que principal par le service d'autorisation installé. Le service d'autorisation par défaut sous Windows ou Linux est appelé Object Authority Manager (OAM). Si vous utilisez la méthode d'accès aux objets (OAM), l'identité doit être définie en tant qu'ID utilisateur.

Utilisez l'identité pour accorder à un client ou à une collection de clients le droit de publier ou de s'abonner à des rubriques définies dans IBM MQ. Si un client MQTT s'est abonné à une rubrique, utilisez l'identité pour lui donner l'autorisation de recevoir les publications résultantes.

Il est difficile de gérer un système avec des dizaines de milliers de clients MQTT , chacun nécessitant des droits d'accès individuels. Une solution consiste à définir des identités communes et à associer des clients MQTT individuels à l'une des identités communes. Définissez autant d'identités communes que nécessaire pour définir différentes combinaisons de droits. Une autre solution consiste à écrire votre propre service d'autorisation qui peut traiter plus facilement avec des milliers d'utilisateurs que le système d'exploitation.

Vous pouvez combiner les clients MQTT en identités communes de deux manières, à l'aide de la méthode d'accès aux objets (OAM):

1. Définissez plusieurs canaux de télémétrie, chacun avec un ID utilisateur différent que l'administrateur alloue à l'aide de IBM MQ Explorer. Les clients qui se connectent à l'aide de numéros de port TCP/IP différents sont associés à des canaux de télémétrie différents et se voient attribuer des identités différentes.

2. Définissez un canal de télémétrie unique, mais chaque client doit sélectionner un nom d'utilisateur dans un petit ensemble d'ID utilisateur. L'administrateur configure le canal de télémétrie pour sélectionner le nom d'utilisateur du client comme identité.

Dans cette tâche, l'identité du canal de télémétrie est appelée *mqttUser*, quel que soit le mode de définition. Si les collections de clients utilisent des identités différentes, utilisez plusieurs *mqttUsers*, un pour chaque collection de clients. Comme la tâche utilise la méthode d'accès aux objets (OAM), chaque *mqttUser* doit être un ID utilisateur.

Pourquoi et quand exécuter cette tâche

Dans cette tâche, vous avez le choix entre quatre modèles de contrôle d'accès que vous pouvez adapter à des exigences spécifiques. Les modèles diffèrent dans leur granularité du contrôle d'accès.

- «Aucun contrôle d'accès», à la page 227
- «Contrôle d'accès à granularité grossière», à la page 227
- «Contrôle d'accès à granularité moyenne», à la page 227
- «Contrôle d'accès à granularité fine», à la page 228

Le résultat de ces modèles est d'affecter à *mqttUsers* des ensembles de droits d'accès pour la publication et l'abonnement à IBM MQ, et de recevoir des publications de IBM MQ.

Aucun contrôle d'accès

Les clients MQTT disposent des droits d'administration IBM MQ et peuvent effectuer n'importe quelle action sur n'importe quel objet.

Procédure

1. Créez un ID utilisateur *mqttUser* pour agir en tant qu'identité de tous les clients MQTT .
2. Ajoutez *mqttUser* au groupe *mqm* ; voir [Ajout d'un utilisateur à un groupe sur Windows](#) ou [Ajout d'un utilisateur à un groupe sur Linux](#)

Contrôle d'accès à granularité grossière

Les clients MQTT sont autorisés à publier et à s'abonner et à envoyer des messages aux clients MQTT . Ils n'ont pas le droit d'effectuer d'autres actions ou d'accéder à d'autres objets.

Procédure

1. Créez un ID utilisateur *mqttUser* pour agir en tant qu'identité de tous les clients MQTT .
2. Autorisez *mqttUser* à publier et à s'abonner à toutes les rubriques et à envoyer des publications aux clients MQTT .

```
setmqaut -m qMgr -t topic -n SYSTEM.BASE.TOPIC -p mqttUser -all +pub +sub
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqttUser -all +put
```

Contrôle d'accès à granularité moyenne

Les clients MQTT sont divisés en différents groupes pour publier et s'abonner à différents ensembles de rubriques et pour envoyer des messages aux clients MQTT .

Procédure

1. Créez plusieurs ID utilisateur, *mqttUsers* et plusieurs rubriques d'administration dans l'arborescence de rubriques de publication / abonnement.
2. Autoriser différents *mqttUsers* à accéder à des rubriques différentes.

```
setmqaut -m qMgr -t topic -n topic1 -p mqttUserA -all +pub +sub
setmqaut -m qMgr -t topic -n topic2 -p mqttUserB -all +pub +sub
```

3. Créez un groupe *mqtt* et ajoutez-y tous les *mqttUsers* .
4. Autorisez *mqtt* à envoyer des rubriques aux clients MQTT .

```
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqtt -all +put
```

Contrôle d'accès à granularité fine

Les clients MQTT sont intégrés à un système de contrôle d'accès existant, qui autorise les groupes à effectuer des actions sur les objets.

Pourquoi et quand exécuter cette tâche

Un ID utilisateur est affecté à un ou plusieurs groupes de système d'exploitation en fonction des autorisations requises. Si les applications IBM MQ publient et s'abonnent au même espace de sujet que les clients MQTT , utilisez ce modèle. Les groupes sont appelés Publish X, Subscribe Y et mqtt

Publish X

Les membres des groupes Publish X peuvent publier dans *topicX*.

Subscribe Y

Les membres des groupes Subscribe Y peuvent s'abonner à *topicY*.

mqtt

Les membres du groupe *mqtt* peuvent envoyer des publications aux clients MQTT .

Procédure

1. Créez plusieurs groupes, Publish X et Subscribe Y , qui sont alloués à plusieurs rubriques d'administration dans l'arborescence de rubriques de publication / abonnement.
2. Créez un groupe mqtt.
3. Créez plusieurs ID utilisateur, *mqttUsers*, et ajoutez les utilisateurs à n'importe quel groupe, en fonction de ce qu'ils sont autorisés à faire.
4. Autorisez différents groupes Publish X et Subscribe X à accéder à des rubriques différentes et autorisez le groupe *mqtt* à envoyer des messages aux clients MQTT .

```
setmqaut -m qMgr -t topic -n topic1 -p Publish X -all +pub  
setmqaut -m qMgr -t topic -n topic1 -p Subscribe X -all +pub +sub  
setmqaut -m qMgr -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p mqtt -all +put
```

Windows

Linux

AIX

MQTT Authentification du client à l'aide d'un mot de passe

Authentifiez le Nom d'utilisateur à l'aide du mot de passe du client. Vous pouvez authentifier le client en utilisant une identité différente de celle utilisée pour autoriser le client à diffuser des publications et à s'abonner aux sujets.

Le service de télémétrie (MQXR) utilise JAAS pour authentifier le Nom d'utilisateur du client. JAAS utilise le mot de passe fourni par le client MQTT .

L'administrateur IBM MQ décide si le Nom d'utilisateur doit être authentifié ou non, en configurant le canal MQTT auquel se connecte un client. Les clients peuvent se voir attribuer des canaux différents et chaque canal peut être configuré pour authentifier ses clients de différentes façons. A l'aide de JAAS, vous pouvez décider quelles sont les méthodes qui doivent authentifier le client, et celles qui peuvent authentifier le client.

Le choix de l'identité pour l'authentification n'affecte pas le choix de l'identité pour l'autorisation. Vous pouvez définir une identité commune pour l'autorisation à des fins d'administration, mais authentifier chaque utilisateur afin qu'il puisse utiliser cette identité. La procédure suivante met en évidence les étapes permettant d'authentifier des utilisateurs individuels pour l'utilisation d'une identité commune :

1. L'administrateur IBM MQ définit l'identité du canal MQTT sur n'importe quel nom, tel que MQTTCClientUser, à l'aide de IBM MQ Explorer.
2. L'administrateur IBM MQ autorise MQTTCClient à diffuser des publications et à s'abonner à n'importe quelle rubrique :

```
setmqaut -m QM1 -t q -n SYSTEM.MQTT.TRANSMIT.QUEUE -p MQTTCClient -all +put  
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p MQTTCClient -all +pub +sub
```

3. Le développeur d'applications client MQTT crée un objet MqttConnectOptions et définit Nom d'utilisateur et Mot de passe avant de se connecter au serveur.
4. Le développeur de sécurité crée un JAAS LoginModule pour authentifier le nom d'utilisateur avec le mot de passe et l'inclut dans le fichier de configuration JAAS .
5. L'administrateur IBM MQ configure le canal MQTT pour authentifier le Username du client à l'aide de JAAS.

Windows

Linux

AIX

Authentification du client MQTT à l'aide de TLS

Les connexions entre le client MQTT et le gestionnaire de files d'attente sont toujours initiées par le client MQTT. Le client MQTT est toujours le client SSL. L'authentification de client du serveur et l'authentification de serveur du client MQTT sont toutes les deux facultatives.

En fournissant au client un certificat numérique signé privé, vous pouvez authentifier le client MQTT auprès de WebSphere MQ. L'administrateur WebSphere peut forcer les clients MQTT à s'authentifier sur le gestionnaire de files d'attente à l'aide de TLS. Vous pouvez uniquement demander l'authentification de client comme faisant partie d'une authentification mutuelle.

Comme alternative à l'utilisation de SSL, certains types de réseaux privés virtuels (VPN), tels que IPsec, authentifient les noeuds finaux d'une connexion TCP/IP. Le réseau privé virtuel VPN chiffre chaque paquet transitant sur le réseau. Une fois qu'une telle connexion VPN est établie, vous avez établi un réseau sécurisé. Vous pouvez connecter les clients MQTT aux canaux de télémétrie avec TCP/IP sur le réseau VPN.

L'authentification de client à l'aide de TLS suppose que le client possède une clé confidentielle. La clé confidentielle est la clé privée du client dans le cas d'un certificat autosigné ou une clé fournie par une autorité de certification. La clé permet de signer le certificat numérique du client. Toute personne en possession de la clé publique correspondante peut vérifier le certificat numérique. Les certificats peuvent être accrédités, ou s'ils font partie d'une chaîne, il est possible de remonter par l'intermédiaire de la chaîne jusqu'au certificat racine accrédité. L'opération de vérification du client envoie tous les certificats de la chaîne fournie par le client au serveur. Le serveur vérifie la chaîne de certificats jusqu'à ce qu'il trouve un certificat digne de confiance. Il s'agit soit du certificat public généré par un certificat autosigné, soit d'un certificat racine généralement émis par une autorité de certification. Au cours de l'étape finale facultative le certificat de confiance peut être comparé à une liste de révocation de certificats "en direct".

Le certificat de confiance peut être émis par une autorité de certification et inclus dans l'espace de stockage de certificats de JRE. Il peut s'agir d'un certificat autosigné ou de tout certificat qui a été ajouté au magasin de clés de canal de télémétrie en tant que certificat digne de confiance.

Remarque : Le canal de télémétrie dispose d'un magasin servant à la fois de magasin de clés et de magasin de clés de confiance, qui contient les clés privées d'accès à un ou plusieurs canaux de télémétrie et les certificats publics nécessaires pour authentifier les clients. Comme un canal SSL doit être associé à un magasin de clés et que ce magasin est le même fichier que le magasin de clés de confiance du canal, l'espace de stockage des certificats du JRE n'est jamais référencé. Il en résulte que si l'authentification d'un client requiert un certificat racine de l'autorité de certification, vous devez placer ce certificat dans le magasin de clés du canal, même si le certificat racine de l'autorité de certification est déjà présent dans l'espace de stockage des certificats du JRE. L'espace de stockage des certificats du JRE n'est jamais référencé.

Ayez à l'esprit les menaces que l'authentification de client peut déjouer et le rôle du client et du serveur pour faire échouer ces menaces. L'authentification du certificat client en soi ne suffit pas à empêcher l'accès non autorisé à un système. Si quelqu'un d'autre a pris la possession du dispositif client, ce dernier

ne s'exécute pas nécessairement avec les droits du détenteur du certificat. Ne faites jamais confiance à une protection unique contre des attaques non désirées. Utilisez au moins une approche de type authentification à deux facteurs et ajoutez en supplément à la possession d'un certificat la connaissance d'informations privées. Par exemple, utilisez JAAS, et authentifiez le client à l'aide d'un mot de passe émis par le serveur.

La principale menace à laquelle est confronté le certificat client est de se trouver aux mains de quelqu'un de mal intentionné. Le certificat se trouve dans un magasin de clés protégé par mot de passe au niveau du client. Comment se retrouve-t-il dans le magasin de clés ? Comment le client MQTT parvient-il à se procurer le mot de passe du magasin de clés ? Quel est le degré de sécurité de la protection par mot de passe ? Les dispositifs de télémétrie sont faciles à supprimer et peuvent donc être piratés en privé. Le dispositif matériel peut-il être imperméable aux actes de malveillance ? La distribution et la protection de certificat côté client sont reconnues comme un acte difficile, on parle alors de problème gestion de clés.

La menace secondaire est le cas où le dispositif est mal utilisé pour accéder au serveur de manière non intentionnelle. Par exemple, si l'application MQTT est trafiquée il est possible d'utiliser une faiblesse dans la configuration du serveur à l'aide de l'identité du client authentifié.

Pour authentifier un client MQTT à l'aide de SSL, configurez le canal de télémétrie et le client.

Concepts associés

[«Configuration du canal de télémétrie pour l'authentification du client MQTT à l'aide de TLS»](#), à la page 230

L'administrateur IBM MQ configure les canaux de télémétrie sur le serveur. Chaque canal est configuré de manière à accepter une connexion TCP/IP sur un numéro de port différent. Les canaux TLS sont configurés avec accès au fichier de clés protégé par phrase passe. Si un canal TLS est défini sans phrase passe ou fichier de clés, le canal n'accepte pas de connexion TLS.

Information associée

[Configuration du client MQTT pour l'authentification du client à l'aide de TLS](#)

Configuration du canal de télémétrie pour l'authentification du client MQTT à l'aide de TLS

L'administrateur IBM MQ configure les canaux de télémétrie sur le serveur. Chaque canal est configuré de manière à accepter une connexion TCP/IP sur un numéro de port différent. Les canaux TLS sont configurés avec accès au fichier de clés protégé par phrase passe. Si un canal TLS est défini sans phrase passe ou fichier de clés, le canal n'accepte pas de connexion TLS.

Définissez la propriété `com.ibm.mq.MQTT.ClientAuth` d'un canal de télémétrie TLS sur `REQUIRED` pour forcer tous les clients à se connecter sur ce canal afin de fournir la preuve qu'ils ont vérifié les certificats numériques. Les certificats client sont authentifiés à l'aide des certificats des autorités de certification, ce qui permet d'obtenir un certificat racine digne de confiance. Si le certificat client est autosigné ou signé par un certificat provenant d'une autorité de certification, les certificats signés publiquement du client ou de l'autorité de certification doivent être stockés de manière sécurisée sur le serveur.

Placez le certificat client signé publiquement ou le certificat de l'autorité de certification dans le magasin de clés du canal de télémétrie. Sur le serveur, les certificats signés publiquement sont stockés dans le même fichier de clés que les certificats signés en privé, plutôt que dans un magasin de clés de confiance distinct.

Le serveur vérifie la signature des certificats client qu'il envoie à l'aide de tous les certificats publics et des suites de chiffrement dont il dispose. Le serveur vérifie la chaîne de clés. Le gestionnaire de files d'attente peut être configuré pour tester le certificat par rapport à la liste de révocation de certificat. La propriété de liste de noms de révocation du gestionnaire de files d'attente est `SSLCRLNL`.

Si l'un des certificats envoyés par un client est vérifié par un certificat dans le magasin de clés du serveur, le client est authentifié.

L'administrateur IBM MQ peut configurer le même canal de télémétrie pour utiliser JAAS afin de vérifier le `UserName` ou `ClientIdentifier` du client avec le mot de passe du client.

Vous pouvez utiliser le même magasin de clés pour plusieurs canaux de télémétrie.

La vérification d'au moins un certificat numérique dans le magasin de clés du client protégé par mot de passe sur le périphérique authentifie le client auprès du serveur. Le certificat numérique est utilisé uniquement pour l'authentification par IBM MQ. Il n'est pas utilisé pour vérifier l'adresse TCP/IP du client, ni pour définir l'identité du client pour l'autorisation ou la comptabilité. L'identité du client adoptée par le serveur est `Username` ou `ClientIdentifier` du client, ou une identité créée par l'administrateur IBM MQ.

Vous pouvez également utiliser des suites de chiffrement TLS pour l'authentification du client. Si vous envisagez d'utiliser des suites de chiffrement SHA-2, voir [«Configuration système requise pour utiliser les suites de chiffrement SHA-2 avec les canaux MQTT»](#), à la page 232.

Concepts associés

[«Configuration de canal de télémétrie pour l'authentification de canal à l'aide de TLS»](#), à la page 232

L'administrateur IBM MQ configure les canaux de télémétrie sur le serveur. Chaque canal est configuré de manière à accepter une connexion TCP/IP sur un numéro de port différent. Les canaux TLS sont configurés avec accès au fichier de clés protégé par phrase passe. Si un canal TLS est défini sans phrase passe ou fichier de clés, le canal n'accepte pas de connexion TLS.

Information associée

[DEFINE CHANNEL \(MQTT\)](#)

[ALTER CHANNEL \(MQTT\)](#)

[CipherSpecs et CipherSuites](#)

Authentification de canal de télémétrie à l'aide de TLS

Les connexions entre le client MQTT et le gestionnaire de files d'attente sont toujours initiées par le client MQTT. Le client MQTT est toujours le client SSL. L'authentification de client du serveur et l'authentification de serveur du client MQTT sont toutes les deux facultatives.

Le client tente toujours d'authentifier le serveur, sauf si le client est configuré pour utiliser un `CipherSpec` qui prend en charge la connexion anonyme. Si l'authentification échoue, la connexion n'est pas établie.

Comme alternative à l'utilisation de SSL, certains types de réseaux privés virtuels (VPN), tels que IPsec, authentifient les noeuds finaux d'une connexion TCP/IP. Le réseau privé virtuel VPN chiffre chaque paquet transitant sur le réseau. Une fois qu'une telle connexion VPN est établie, vous avez établi un réseau sécurisé. Vous pouvez connecter les clients MQTT aux canaux de télémétrie avec TCP/IP sur le réseau VPN.

L'authentification de serveur à l'aide de SSL permet d'authentifier le serveur auquel vous êtes sur le point d'envoyer des informations confidentielles. Le client effectue les vérifications correspondant aux certificats envoyés à partir du serveur, par rapport aux certificats placés dans son magasin de clés de confiance ou dans son magasin JRE `cacerts`.

Le magasin de certificats JRE est un fichier JKS, `cacerts`. Il se trouve dans `JRE InstallPath\lib\security\`. Il est installé avec le mot de passe par défaut `changeit`. Vous pouvez stocker les certificats que vous accédez dans l'espace de stockage des certificats JRE ou dans le magasin de clés de confiance. Vous ne pouvez pas utiliser les deux magasins. Utilisez le fichier de clés certifiées si vous souhaitez conserver les certificats publics et les certificats utilisés par d'autres applications Java à des emplacements distincts. Utilisez l'espace de stockage des certificats JRE si vous souhaitez utiliser un certificat commun pour toutes les applications Java qui s'exécutent sur le client. Si vous décidez d'utiliser cet espace de stockage, vérifiez les certificats qu'il contient afin d'être sûr que vous les accédez.

Vous pouvez modifier la configuration JSSE en indiquant un autre fournisseur d'accréditation. Vous pouvez personnaliser un fournisseur d'accréditation pour effectuer différentes vérifications sur un certificat. Dans certains environnements OGSi qui ont utilisé le client MQTT, l'environnement fournit un fournisseur d'accréditation différent.

Pour authentifier un canal de télémétrie à l'aide de TLS, configurez le serveur et le client.

Windows Linux AIX Configuration de canal de télémétrie pour l'authentification de canal à l'aide de TLS

L'administrateur IBM MQ configure les canaux de télémétrie sur le serveur. Chaque canal est configuré de manière à accepter une connexion TCP/IP sur un numéro de port différent. Les canaux TLS sont configurés avec accès au fichier de clés protégé par phrase passe. Si un canal TLS est défini sans phrase passe ou fichier de clés, le canal n'accepte pas de connexion TLS.

Stockez le certificat numérique du serveur, signé avec sa clé privée, dans le magasin de clés que le canal de télémétrie va utiliser sur le serveur. Stockez tous les certificats de sa chaîne de clés dans le magasin de clés, si vous souhaitez transmettre la chaîne de clés au client. Configurez le canal de télémétrie à l'aide de l'explorateur IBM MQ pour utiliser TLS. Indiquez le chemin d'accès au magasin de clés et la phrase passe pour accéder au magasin de clés. Si vous ne définissez pas le numéro de port TCP/IP du canal, le numéro de port du canal de télémétrie TLS est par défaut 8883.

Vous pouvez également utiliser des suites de chiffrement TLS pour l'authentification de canal. Si vous envisagez d'utiliser des suites de chiffrement SHA-2, voir [«Configuration système requise pour utiliser les suites de chiffrement SHA-2 avec les canaux MQTT»](#), à la page 232.

Concepts associés

[«Configuration du canal de télémétrie pour l'authentification du client MQTT à l'aide de TLS»](#), à la page 230

L'administrateur IBM MQ configure les canaux de télémétrie sur le serveur. Chaque canal est configuré de manière à accepter une connexion TCP/IP sur un numéro de port différent. Les canaux TLS sont configurés avec accès au fichier de clés protégé par phrase passe. Si un canal TLS est défini sans phrase passe ou fichier de clés, le canal n'accepte pas de connexion TLS.

Information associée

[DEFINE CHANNEL \(MQTT\)](#)

[ALTER CHANNEL \(MQTT\)](#)

[CipherSpecs et CipherSuites](#)

Windows Linux AIX Configuration système requise pour utiliser les suites de chiffrement SHA-2 avec les canaux MQTT

Si vous utilisez une version de Java qui prend en charge les suites de chiffrement SHA-2, vous pouvez utiliser ces suites pour sécuriser vos canaux MQTT (télémétrie) et vos applications client.

Pour IBM MQ 8.0, qui inclut le service de télémétrie (MQXR), la version minimale de Java est Java 7 à partir de IBM, SR6. Les suites de chiffrement SHA-2 sont prises en charge par défaut dans Java 7 à partir de IBM, SR4. Vous pouvez donc utiliser des suites de chiffrement SHA-2 avec le service de télémétrie (MQXR) pour sécuriser vos canaux MQTT (télémétrie).

Si vous exécutez un client MQTT avec un JRE différent, vous devez vous assurer qu'il prend également en charge les suites de chiffrement SHA-2.

Concepts associés

[«Configuration de canal de télémétrie pour l'authentification de canal à l'aide de TLS»](#), à la page 232

L'administrateur IBM MQ configure les canaux de télémétrie sur le serveur. Chaque canal est configuré de manière à accepter une connexion TCP/IP sur un numéro de port différent. Les canaux TLS sont configurés avec accès au fichier de clés protégé par phrase passe. Si un canal TLS est défini sans phrase passe ou fichier de clés, le canal n'accepte pas de connexion TLS.

Information associée

[Service de télémétrie \(MQXR\)](#)

[DEFINE CHANNEL \(MQTT\)](#)

[ALTER CHANNEL \(MQTT\)](#)

Confidentialité de la publication sur les canaux de télémétrie

La confidentialité des publications MQTT envoyées dans les deux sens sur les canaux de télémétrie est sécurisée à l'aide de TLS pour chiffrer les transmissions sur la connexion.

Les clients MQTT qui se connectent aux canaux de télémétrie utilisent TLS pour sécuriser la confidentialité des publications transmises sur le canal à l'aide de la cryptographie à clé symétrique. Etant donné que les noeuds finaux ne sont pas authentifiés, vous ne pouvez pas accréditer le chiffrement de canal seul. Combinez la sécurisation de la confidentialité avec l'authentification du serveur ou l'authentification mutuelle.

Comme alternative à l'utilisation de SSL, certains types de réseaux privés virtuels (VPN), tels que IPsec, authentifient les noeuds finaux d'une connexion TCP/IP. Le réseau privé virtuel VPN chiffre chaque paquet transitant sur le réseau. Une fois qu'une telle connexion VPN est établie, vous avez établi un réseau sécurisé. Vous pouvez connecter les clients MQTT aux canaux de télémétrie avec TCP/IP sur le réseau VPN.

Pour une configuration classique, qui chiffre le canal et authentifie le serveur, consultez la section «Authentification de canal de télémétrie à l'aide de TLS», à la page 231.

Le chiffrement des connexions TLS sans authentification du serveur expose la connexion aux attaques man-in-the-middle. Bien que les informations que vous échangez soient protégées contre les écoutes clandestines, vous ne savez pas avec qui vous réalisez les échanges. A moins que vous ne contrôliez le réseau, vous êtes exposé à ce que quelqu'un intercepte vos transmissions IP, en se faisant passer pour un noeud final.

Vous pouvez créer une connexion TLS chiffrée, sans authentifier le serveur, à l'aide d'un CipherSpec d'échange de clés Diffie-Hellman qui prend en charge le protocole TLS anonyme. Le secret maître, partagé entre le client et le serveur et utilisé pour chiffrer les transmissions TLS, est établi sans échanger de certificat de serveur signé de manière privée.

Etant donné que les connexions anonymes ne sont pas sécurisées, la plupart des implémentations TLS n'utilisent pas par défaut les CipherSpecs anonymes. Si une demande client de connexion TLS est acceptée par un canal de télémétrie, le canal doit disposer d'un magasin de clés protégé par une phrase passe. Par défaut, les implémentations TLS n'utilisant pas de CipherSpecs anonyme, le magasin de clés doit contenir un certificat signé de manière privée que le client peut authentifier.

Si vous utilisez des spécifications CipherSpecs anonymes, le magasin de clés du serveur doit exister, mais il ne doit pas forcément contenir des certificats signés de manière privée.

Une autre façon d'établir une connexion cryptée consiste à remplacer le fournisseur d'accréditation côté client par votre propre implémentation. Votre fournisseur d'accréditation n'effectuerait pas l'authentification du certificat serveur, mais la connexion serait cryptée.



Avertissement : Lorsque vous utilisez TLS avec MQTT, vous pouvez utiliser des messages volumineux, mais cela peut avoir un impact possible sur les performances. MQTT est optimisé pour le traitement des messages de petite taille (généralement entre 1KB et 1MB).

Configuration TLS des clients MQTT Java et des canaux de télémétrie

Configurez TLS pour authentifier le canal de télémétrie et le client MQTT Java, et chiffrez le transfert des messages entre eux. MQTT Les clients Java utilisent JSSE (Java Secure Socket Extension) pour connecter les canaux de télémétrie à l'aide de TLS. Comme alternative à l'utilisation de SSL, certains types de réseaux privés virtuels (VPN), tels que IPsec, authentifient les noeuds finaux d'une connexion TCP/IP. Le réseau privé virtuel VPN chiffre chaque paquet transitant sur le réseau. Une fois qu'une telle connexion VPN est établie, vous avez établi un réseau sécurisé. Vous pouvez connecter les clients MQTT aux canaux de télémétrie avec TCP/IP sur le réseau VPN.

Vous pouvez configurer la connexion entre un client Java MQTT et un canal de télémétrie pour utiliser le protocole TLS sur TCP/IP. Ce qui est sécurisé dépend de la façon dont vous configurez TLS pour utiliser

JSSE. En partant de la configuration la plus sécurisée, vous pouvez configurer trois niveaux différents de sécurité :

1. Autorisez uniquement les clients MQTT dignes de confiance à se connecter. Connectez un client MQTT uniquement à un canal de télémétrie sécurisé. Chiffrez les messages entre le client et le gestionnaire de files d'attente ; voir «Authentification du client MQTT à l'aide de TLS», à la page 229
2. Connectez un client MQTT uniquement à un canal de télémétrie sécurisé. Chiffrez les messages entre le client et le gestionnaire de files d'attente ; voir «Authentification de canal de télémétrie à l'aide de TLS», à la page 231.
3. Chiffrez les messages entre le client et le gestionnaire de files d'attente ; voir «Confidentialité de la publication sur les canaux de télémétrie», à la page 233.

Paramètres de configuration JSSE

Modifiez les paramètres JSSE pour modifier la manière dont une connexion TLS est configurée. Les paramètres de configuration JSSE sont composés de trois ensembles :

1. [Canal MQ Telemetry](#)
2. [clientMQTT Java](#)
3. [JRE](#)

Configurez les paramètres du canal de télémétrie à l'aide d'IBM MQ Explorer. Définissez les paramètres du client MQTT Java dans l'attribut `MqttConnectionOptions.SSLProperties`. Modifiez les paramètres de sécurité en éditant les fichiers dans le répertoire de sécurité de JRE à la fois sur le client et sur le serveur.

MQ Telemetry canal

Définissez tous les paramètres TLS du canal de télémétrie à l'aide de IBM MQ Explorer.

ChannelName

`ChannelName` est un paramètre obligatoire sur tous les canaux.

Le nom de canal identifie le canal associé à un numéro de port particulier. Nommez des canaux pour vous aider à administrer des ensembles de clients MQTT .

PortNumber

`PortNumber` est un paramètre facultatif sur tous les canaux. La valeur par défaut est 1883 pour les canaux TCP et 8883 pour les canaux TLS.

Numéro de port TCP/IP associé à ce canal. Les clients MQTT sont connectés à un canal en spécifiant le port défini pour le canal. Si le canal possède des propriétés TLS, le client doit se connecter à l'aide du protocole TLS ; par exemple:

```
MQTTClient mqttClient = new MqttClient( "ssl://www.example.org:8884", "clientId1");
mqttClient.connect();
```

KeyFileName

`KeyFileName` est un paramètre obligatoire pour les canaux TLS. Il doit être omis pour les canaux TCP.

`KeyFileName` est le chemin d'accès au magasin de clés Java contenant les certificats numériques que vous fournissez. Sur le serveur, utilisez les types de magasin de clés JKS, JCEKS ou PKCS12.

Identifiez le type de magasin de clés à l'aide de l'une des extensions de fichier suivantes :

- .jks
- .jceks
- .p12
- .pkcs12

Un magasin de clés associé à une autre extension est censé être un magasin JKS.

Vous pouvez combiner un magasin de clés d'un type donné se trouvant sur le serveur avec d'autres types de magasins de clés se trouvant sur le client.

Placez le certificat privé du serveur dans le magasin de clés. Le certificat est connu sous le nom de certificat serveur. Ce certificat peut être autosigné ou faire partie d'une chaîne de certificats signée par une autorité de signature.

Si vous utilisez une chaîne de certificats, placez les certificats associés dans le magasin de clés du serveur.

Le certificat du serveur et tous les certificats de la chaîne de certificats sont envoyés aux clients afin d'authentifier l'identité du serveur.

Si vous avez attribué à `ClientAuth` la valeur `Required`, le magasin de clés doit contenir tous les certificats nécessaires à l'authentification du client. Le client envoie un certificat autosigné, ou une chaîne de certificats, et est authentifié par la première vérification de cet élément par rapport à un certificat se trouvant dans le magasin de clés. Lorsque vous utilisez une chaîne de certificats, un certificat peut vérifier plusieurs clients, même s'ils sont émis avec des certificats client différents.

PassPhrase

`PassPhrase` est un paramètre obligatoire pour les canaux TLS. Il doit être omis pour les canaux TCP.

La phrase passe est utilisée pour protéger le magasin de clés.

ClientAuth

`ClientAuth` est un paramètre TLS facultatif. Par défaut, aucune authentification de client n'est effectuée. Il doit être omis pour les canaux TCP.

Définissez `ClientAuth` si vous voulez que le service de télémétrie (MQXR) authentifie le client avant de lui permettre de se connecter au canal de télémétrie.

Si vous définissez `ClientAuth`, le client doit se connecter au serveur à l'aide de TLS et authentifier le serveur. En réponse à la définition de `ClientAuth`, le client envoie son certificat numérique au serveur et tout autre certificat dans son magasin de clés. Son certificat numérique est connu sous le nom de certificat client. Ces certificats sont authentifiés par rapport aux certificats se trouvant dans le magasin de clés du canal et dans le magasin `cacerts`.

CipherSuite

`CipherSuite` est un paramètre TLS facultatif. Par défaut, il essaie toutes les spécifications `CipherSpecs` activées. Il doit être omis pour les canaux TCP.

Si vous souhaitez utiliser un `CipherSpec` particulier, définissez `CipherSuite` sur le nom du `CipherSpec` qui doit être utilisé pour établir la connexion TLS.

Le service de télémétrie et le client MQTT négocient un `CipherSpec` commun à partir de tous les `CipherSpecs` activés à chaque extrémité. Si un `CipherSpec` spécifique est indiqué à l'une ou aux deux extrémités, il doit correspondre au `CipherSpec` de l'autre extrémité.

Installez les chiffrements supplémentaires en ajoutant des fournisseurs supplémentaires à JSSE.

FIPS (Federal Information Processing Standards)

`FIPS` est un paramètre facultatif. Par défaut, il n'est pas défini.

Utilisez le panneau de propriété du gestionnaire de files d'attente ou `runmqsc`, pour définir `SSLFIPS`. `SSLFIPS` Indique si seuls les algorithmes certifiés pour FIPS doivent être utilisés.

Revocation namelist

`Revocation namelist` est un paramètre facultatif. Par défaut, il n'est pas défini.

Utilisez le panneau de propriété du gestionnaire de files d'attente ou **runmqsc**, pour définir SSLCRLNL. SSLCRLNL indique une liste de noms d'objet d'informations d'authentification utilisés pour fournir des emplacements de révocation de certificat.

Aucun autre paramètre de gestionnaire de files d'attente qui définit les propriétés TLS n'est utilisé.

Client MQTTJava

Définissez les propriétés TLS du client Java dans `MqttConnectionOptions.SSLProperties`; par exemple:

```
java.util.Properties sslClientProperties = new Properties();
sslClientProperties.setProperty("com.ibm.ssl.keyStoreType", "JKS");
com.ibm.micro.client.mqttv3.MqttConnectOptions conOptions = new MqttConnectOptions();
conOptions.setSSLProperties(sslClientProperties);
```

Les noms et les valeurs de propriétés spécifiques sont décrits dans la classe `MqttConnectOptions`. Pour des liens vers la documentation de l'API client pour les bibliothèques client MQTT, voir [MQTT client programming reference](#).

Protocole

`Protocol` est facultatif.

Le protocole est sélectionné en négociation avec le serveur de télémétrie. Si vous avez besoin d'un protocole spécifique, vous pouvez en sélectionner un. Si le serveur de télémétrie ne prend pas en charge le protocole, la connexion échoue.

ContextProvider

`ContextProvider` est facultatif.

KeyStore

`KeyStore` est facultatif. Configurez-le si `ClientAuth` est défini au niveau du serveur pour forcer l'authentification du client.

Placez le certificat numérique du client, signé à l'aide de la clé privée, dans le magasin de clés. Spécifiez le chemin d'accès et le mot de passe du magasin de clés. Le type et le fournisseur sont facultatifs. JKS est le type par défaut et IBMJCE est le fournisseur par défaut.

Indiquez un fournisseur de magasin de clés pour référencer une classe qui ajoute un nouveau fournisseur de magasin de clés. Transmettez le nom de l'algorithme utilisé par le fournisseur de magasin de clés pour instancier `KeyManagerFactory` en définissant le nom du gestionnaire de clés.

TrustStore

`TrustStore` est facultatif. Vous pouvez placer tous les certificats que vous accédez dans le magasin `cacerts` de JRE.

Configurez le magasin de clés de confiance si vous souhaitez disposer d'un fichier de ce type distinct pour le client. Vous pouvez ne pas configurer le magasin de clés de confiance si le serveur utilise un certificat émis par une autorité de certification connue dont le certificat racine est déjà stocké dans `cacerts`.

Ajoutez le certificat signé publiquement du serveur ou du certificat racine au magasin de clés de confiance, et indiquez le chemin d'accès et le mot de passe du magasin de clés de confiance. JKS est le type par défaut et IBMJCE est le fournisseur par défaut.

Indiquez un fournisseur de magasin de clés de confiance pour référencer une classe qui ajoute un nouveau fournisseur de magasin de clés de confiance. Transmettez le nom de l'algorithme utilisé par le fournisseur de magasin de clés de confiance pour instancier `TrustManagerFactory` en définissant le nom du gestionnaire d'accréditation.

JRE

D'autres aspects de la sécurité Java qui affectent le comportement de TLS sur le client et le serveur sont configurés dans l'environnement d'exécution Java. Les fichiers de configuration sous Windows se trouvent dans *Java Installation Directory\jre\lib\security*. Si vous utilisez le JRE livré avec IBM MQ, le chemin est celui qui est indiqué dans le tableau suivant :

| <i>Tableau 16. Chemins d'accès aux fichiers par plateforme pour les fichiers de configuration TLS de l'environnement d'exécution Java</i> | |
|---|---|
| Plateforme | Chemin d'accès au fichier |
| Windows | <i>WMQ Installation Directory\java\jre\lib\security</i> |
| UNIX and Linux plateformes | <i>WMQ Installation Directory/java/jre64/jre/lib/security</i> |

Autorités de certification reconnues

Le fichier `cacerts` contient les certificats racine des autorités de certification reconnues. Le fichier `cacerts` est utilisé par défaut, sauf si vous définissez un autre magasin de clés de confiance. Si vous utilisez le fichier `cacerts` ou que vous n'indiquez aucun magasin de clés de confiance, vous devez vérifier et éditer la liste des signataires du fichier `cacerts` pour remplir les conditions requises en matière de sécurité.

Vous pouvez ouvrir `cacerts` à l'aide de la commande IBM MQ `strmqikm` qui exécute l'utilitaire IBM Key Management. Ouvrez le fichier `cacerts` en tant que fichier JKS avec le mot de passe `changeit`. Modifiez le mot de passe afin de sécuriser le fichier.

Configuration des classes de sécurité

Utilisez le fichier `java.security` pour enregistrer les fournisseurs de sécurité supplémentaires et d'autres propriétés de sécurité par défaut.

Droits d'accès

Utilisez le fichier `java.policy` pour modifier les droits d'accès accordés aux ressources. `javaws.policy` accorde des droits à `javaws.jar`

Puissance de chiffrement

Certains JRE sont livrés avec une puissance de chiffrement réduite. Si vous ne pouvez pas importer les clés dans les magasins de clés, la réduction de la puissance de chiffrement peut en être la cause. Essayez de démarrer **ikeman** à l'aide de la commande **strmqikm** ou téléchargez des fichiers de juridiction forte, mais limitée, à partir des kits de développement [IBM](#), des [informations de sécurité](#).

Important : Votre pays d'origine est peut-être assujéti à des restrictions relatives à l'importation, la possession, l'utilisation ou la réexportation de logiciel de chiffrement vers un autre pas. Avant de télécharger ou d'utiliser des fichiers de règles sans restriction, vous devez vérifier les lois applicables dans votre pays. Vérifiez ses règlements et sa politique en matière d'importation, de possession, d'utilisation et de réexportation de logiciel de chiffrement afin de déterminer si cette action est autorisée.

Modifiez le fournisseur d'accréditation pour permettre au client de se connecter à n'importe quel serveur.

L'exemple montre comment ajouter un fournisseur d'accréditation et le référencer à partir du code client MQTT. Dans l'exemple aucune authentification du client ou du serveur n'est effectuée. La connexion TLS résultante est chiffrée sans être authentifiée.

Le fragment de code dans [Figure 26](#), à la page 238 définit le fournisseur de relations de confiance et le gestionnaire de relations de confiance `AcceptAllProviders` pour le client MQTT.

```

java.security.Security.addProvider(new AcceptAllProvider());
java.util.Properties sslClientProperties = new Properties();
sslClientProperties.setProperty("com.ibm.ssl.trustManager", "TrustAllCertificates");
sslClientProperties.setProperty("com.ibm.ssl.trustStoreProvider", "AcceptAllProvider");
conOptions.setSSLProperties(sslClientProperties);

```

Figure 26. Fragment de code du client MQTT

```

package com.ibm.mq.id;
public class AcceptAllProvider extends java.security.Provider {
private static final long serialVersionUID = 1L;
public AcceptAllProvider() {
super("AcceptAllProvider", 1.0, "Trust all X509 certificates");
put("TrustManagerFactory.TrustAllCertificates",
AcceptAllTrustManagerFactory.class.getName());
}
}

```

Figure 27. AcceptAllProvider.java

```

protected static class AcceptAllTrustManagerFactory extends
javax.net.ssl.TrustManagerFactorySpi {
public AcceptAllTrustManagerFactory() {}
protected void engineInit(java.security.KeyStore keystore) {}
protected void engineInit(
javax.net.ssl.ManagerFactoryParameters parameters) {}
protected javax.net.ssl.TrustManager[] engineGetTrustManagers() {
return new javax.net.ssl.TrustManager[] { new AcceptAllX509TrustManager() };
}
}

```

Figure 28. AcceptAllTrustManagerFactory.java

```

protected static class AcceptAllX509TrustManager implements
javax.net.ssl.X509TrustManager {
public void checkClientTrusted(
java.security.cert.X509Certificate[] certificateChain,
String authType) throws java.security.cert.CertificateException {
report("Client authtype=" + authType);
for (java.security.cert.X509Certificate certificate : certificateChain) {
report("Accepting:" + certificate);
}
}
public void checkServerTrusted(
java.security.cert.X509Certificate[] certificateChain,
String authType) throws java.security.cert.CertificateException {
report("Server authtype=" + authType);
for (java.security.cert.X509Certificate certificate : certificateChain) {
report("Accepting:" + certificate);
}
}
public java.security.cert.X509Certificate[] getAcceptedIssuers() {
return new java.security.cert.X509Certificate[0];
}
private static void report(String string) {
System.out.println(string);
}
}
}

```

Figure 29. AcceptAllX509TrustManager.java

Windows Linux AIX Configuration JAAS du canal de télémétrie

Configurez JAAS pour authentifier le Nom d'utilisateur envoyé par le client.

L'administrateur IBM MQ configure les canaux MQTT qui requièrent l'authentification de client à l'aide de JAAS. Indiquez le nom d'une configuration JAAS pour chaque canal qui doit effectuer une authentification JAAS. Les canaux peuvent tous utiliser la même configuration JAAS configuration, ou utiliser des configurations JAAS différentes. Les configurations sont définies dans *WMQData directory\qmgrs\qMgrName\mqxr\jaas.config*.

Le fichier `jaas.config` est organisé par nom de configuration JAAS. Sous chaque nom de configuration figure une liste de configurations de connexion ; voir [Figure 30](#), à la page 239.

JAAS fournit quatre normes de modules de connexion. Les modules standard NT et UNIX Login sont d'une valeur limitée.

JndiLoginModule

S'authentifie auprès d'un service d'annuaire configuré sous JNDI (Java Naming and Directory Interface).

Krb5LoginModule

Effectue l'authentification à l'aide des protocoles Kerberos.

NTLoginModule

Effectue l'authentification à l'aide des informations de sécurité NT pour l'utilisateur en cours.

UnixLoginModule

S'authentifie à l'aide des informations de sécurité UNIX pour l'utilisateur en cours.

Le problème lié à l'utilisation de `NTLoginModule` ou de `UnixLoginModule` est que le service de télémétrie (MQXR) s'exécute avec l'identité `mqm` et non avec l'identité du canal MQTT. `mqm` est l'identité transmise à `NTLoginModule` ou `UnixLoginModule` pour l'authentification et non pas l'identité du client.

Pour résoudre ce problème, écrivez votre propre module de connexion ou utilisez d'autres modules de connexion standard. Un exemple de fichier `JAASLoginModule.java` est fourni avec MQ Telemetry. Il s'agit d'une implémentation de l'interface `javax.security.auth.spi.LoginModule`. Utilisez-le pour développer votre propre méthode d'authentification.

Toute nouvelle classe `LoginModule` que vous fournissez doit figurer dans le chemin d'accès aux classes du service de télémétrie (MQXR). Ne placez pas vos classes dans les répertoires IBM MQ qui se trouvent dans le chemin d'accès aux classes. Créez vos propres répertoires et définissez le chemin d'accès complet aux classes pour le service de télémétrie (MQXR).

Vous pouvez accroître le chemin d'accès aux classes utilisé par le service de télémétrie (MQXR) en définissant le chemin d'accès aux classes dans le fichier `service.env`. `CLASSPATH` doit être indiqué en majuscules et peut uniquement contenir des littéraux. Vous ne pouvez pas utiliser de variables dans `CLASSPATH` ; par exemple, l'instruction `CLASSPATH=%CLASSPATH%` est incorrecte. Le service de télémétrie (MQXR) définit son propre chemin d'accès aux classes? La variable `CLASSPATH` définie dans le fichier `service.env` est ajoutée à ce dernier.

Le service de télémétrie (MQXR) fournit deux rappels qui renvoient le nom d'utilisateur et le mot de passe pour un client connecté au canal MQTT. Le nom d'utilisateur et le mot de passe sont définis dans l'objet `MqttConnectOptions`. Voir la [Figure 31](#), à la page 240 pour un exemple d'accès au Nom d'utilisateur et au Mot de passe.

Exemples

Modèle de fichier de configuration JAAS avec une configuration nommée, `MQXRConfig`.

```
MQXRConfig {
samples.JAASLoginModule required debug=true;
//com.ibm.security.auth.module.NTLoginModule required;
//com.ibm.security.auth.module.Krb5LoginModule required
//    principal=principal@your_realm
//    useDefaultCcache=TRUE
//    renewTGT=true;
//com.sun.security.auth.module.NTLoginModule required;
//com.sun.security.auth.module.UnixLoginModule required;
//com.sun.security.auth.module.Krb5LoginModule required
//    useTicketCache="true"
//    ticketCache="${user.home}/${}/tickets";
};
```

Figure 30. Modèle de fichier `jaas.config`

Exemple de module de connexion JAAS codé pour recevoir le nom d'utilisateur et le mot de passe fournis par un client MQTT .

```
public boolean login()
throws javax.security.auth.login.LoginException {
    javax.security.auth.callback.Callback[] callbacks =
    new javax.security.auth.callback.Callback[2];
    callbacks[0] = new javax.security.auth.callback.NameCallback("NameCallback");
    callbacks[1] = new javax.security.auth.callback.PasswordCallback(
    "PasswordCallback", false);
    try {
        callbackHandler.handle(callbacks);
        String username = ((javax.security.auth.callback.NameCallback) callbacks[0])
        .getName();
        char[] password = ((javax.security.auth.callback.PasswordCallback) callbacks[1])
        .getPassword();
        // Accept everything.
        if (true) {
            loggedIn = true;
        } else
        throw new javax.security.auth.login.FailedLoginException("Login failed");

        principal= new JAASPrincipal(username);

    } catch (java.io.IOException exception) {
        throw new javax.security.auth.login.LoginException(exception.toString());
    } catch (javax.security.auth.callback.UnsupportedCallbackException exception) {
        throw new javax.security.auth.login.LoginException(exception.toString());
    }

    return loggedIn;
}
```

Figure 31. Modèle de méthode `JAASLoginModule.Login()`

Information associée

Classe `MQXR AuthCallback`

Résolution des problèmes : [Le module de connexion JAAS n'est pas appelé par le service de télémétrie](#)

V 9.0.0 Administration d'IBM MQ Light

Vous pouvez administrer MQ Light à l'aide de IBM MQ Explorer ou d'une ligne de commande. Utilisez l'explorateur pour configurer des canaux et surveiller les clients MQ Light qui sont connectés à IBM MQ. Configurez la sécurité de MQ Light à l'aide de TLS et de JAAS.

Avant de commencer

Pour plus d'informations sur l'installation d'AMQP sur votre plateforme, voir [Choix concernant ce qu'il faut installer](#). Installez le composant AMQP Service à l'aide de la mise à jour du fabricant IBM MQ 8.0.0.4 et non du groupe de correctifs 8.0.0.4 . Vous ne pouvez pas installer le composant AMQP sur une version du gestionnaire de files d'attente antérieure à 8.0.0.4.

Administration à l'aide de IBM MQ Explorer

Utilisez l'explorateur pour configurer les canaux AMQP et surveiller les clients MQ Light qui sont connectés à IBM MQ. Vous pouvez configurer la sécurité de MQ Light à l'aide de TLS et de JAAS.

Administration à l'aide de la ligne de commande

Vous pouvez administrer MQ Light à partir de la ligne de commande à l'aide des commandes IBM MQ [MQSC](#) .

Vous pouvez afficher les différentes ressources IBM MQ utilisées par les clients MQ Light , par exemple les connexions et les abonnements.

Connexions

Lorsque le service AMQP est démarré, de nouvelles connexions sont créées et connectées au gestionnaire de files d'attente. Ce pool de connexions Hconns est utilisé lorsque les clients MQ Light publient des messages. Vous pouvez afficher les connexions Hconns à l'aide de la commande **DISPLAY CONN** . Exemple :

```
DISPLAY CONN(*) TYPE(CONN) WHERE (APPLDESC LK 'WebSphere MQ Advanced Message Queuing Protocol*')
```

Cette commande affiche également les connexions Hconns spécifiques au client. Les connexions Hconns dont l'attribut d'ID client est vide sont les connexions Hconns utilisées dans le pool

Lorsqu'un client MQ Light se connecte à un canal AMQP, un nouveau Hconn est connecté au gestionnaire de files d'attente. Cette Hconn est utilisée pour consommer les messages de manière asynchrone pour les abonnements créés par le client MQ Light . Vous pouvez afficher la valeur Hconn utilisée par un client MQ Light particulier à l'aide de la commande **DISPLAY CONN** . Exemple :

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_abcd1234')
```

Abonnements créés par les clients

Lorsqu'un client MQ Light s'abonne à une rubrique, un nouvel abonnement IBM MQ est créé. Le nom de l'abonnement inclut les informations suivantes:

- Nom du client. Si le client a rejoint un abonnement partagé, le nom du partage est utilisé
- Modèle de rubrique auquel le client s'est abonné
- Préfixe. Le préfixe est `private` si le client a créé un abonnement non partagé ou `share` si le client a rejoint un abonnement partagé

Pour afficher les abonnements utilisés par un client MQ Light particulier, exécutez la commande **DISPLAY SUB** et filtrez sur le préfixe `private` :

```
DISPLAY SUB('/:private:*')
```

Pour afficher les abonnements partagés qui sont utilisés par plusieurs clients, exécutez la commande **DISPLAY SUB** et filtrez sur le préfixe `share` :

```
DISPLAY SUB('/:share:*')
```

Etant donné que les abonnements partagés peuvent être utilisés par plusieurs clients MQ Light , vous pouvez afficher les clients qui consomment actuellement des messages de l'abonnement partagé. Pour ce faire, vous pouvez répertorier les connexions Hconns dont un descripteur est actuellement ouvert dans la file d'attente d'abonnement. Pour afficher les clients qui utilisent actuellement un partage, procédez comme suit:

1. Recherchez le nom de la file d'attente que l'abonnement partagé utilise comme destination. Exemple :

```
DISPLAY SUB('/:private:recv_e298452:public') DEST
5 : DISPLAY SUB('/:private:recv_e298452:public') DEST
AMQ8096: WebSphere MQ subscription inquired.
SUBID(414D5120514D3120202020202020202020707E0A565C2D0020)
SUB(:private:recv_e298452:public)
DEST(SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
```

2. Exécutez la commande **DISPLAY CONN** pour rechercher les descripteurs ouverts sur cette file d'attente:

```
DISPLAY CONN(*) TYPE(HANDLE) WHERE (OBJNAME
EQ SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)
21 : DISPLAY CONN(*) TYPE(HANDLE) WHERE(OBJNAME EQ
SYSTEM.MANAGED.DURABLE.560A7E7020002D5B)

AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(HANDLE)

OBJNAME(SYSTEM.BASE.TOPIC)      OBJTYPE(TOPIC)

OBJNAME(SYSTEM.MANAGED.DURABLE.560A7E7020002961)
OBJTYPE(Queue)
```

3. Pour chacun des descripteurs, affichez l'ID client MQ Light dont le descripteur est ouvert:

```
DISPLAY CONN(707E0A56642B0020) CLIENTID
23 : DISPLAY CONN(707E0A56642B0020) CLIENTID

AMQ8276: Display Connection details.
CONN(707E0A56642B0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
CLIENTID(recv_8f02c9d)
DISPLAY CONN(707E0A565F290020) CLIENTID
24 : DISPLAY CONN(707E0A565F290020) CLIENTID
AMQ8276: Display Connection details.
CONN(707E0A565F290020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
CLIENTID(recv_86d8888)
```

V 9.0.0

MQ Light Identification, autorisation et authentification du client

Comme les autres applications client IBM MQ, vous pouvez sécuriser les connexions AMQP de différentes manières.

Vous pouvez utiliser les fonctions de sécurité suivantes pour sécuriser les connexions AMQP à IBM MQ:

- [Enregistrements d'authentification de canal](#)
- [Authentification de connexion](#)
- Configuration utilisateur MCA de canal
- Définitions des droits d'accès IBM MQ
- [Connectivité TLS](#)

Du point de vue de la sécurité, l'établissement d'une connexion comprend les deux étapes suivantes:

- Décider si la connexion doit se poursuivre
- Choix de l'identité IBM MQ que l'application suppose pour les vérifications ultérieures des droits d'accès

Les informations suivantes décrivent les différentes configurations IBM MQ et les étapes à suivre lorsqu'un client AMQP tente d'établir une connexion. Toutes les configurations IBM MQ n'utilisent pas toutes les étapes décrites. Par exemple, certaines configurations n'utilisent pas TLS pour les connexions à l'intérieur du pare-feu de la société et d'autres utilisent TLS mais n'utilisent pas de certificats client pour l'authentification. De nombreux environnements n'utilisent pas de modules JAAS personnalisés ou personnalisés.

Etablissement d'une connexion

Les étapes suivantes décrivent ce qui se passe lorsqu'une connexion est établie par un client AMQP. Les étapes déterminent si la connexion se poursuit et quelle identité IBM MQ l'application utilise pour les vérifications des droits d'accès:

1. Si le client ouvre une connexion TLS à IBM MQ et fournit un certificat, le gestionnaire de files d'attente tente de valider le certificat client.
2. Si le client fournit des données d'identification par nom d'utilisateur et mot de passe, une trame AMQP SASL est reçue par le gestionnaire de files d'attente et la configuration MQ CONNAUTH est vérifiée.
3. Les règles d'authentification de canal MQ sont vérifiées (par exemple, si l'adresse IP et le nom distinctif du certificat TLS sont valides)
4. Le canal MCAUSER est activé, sauf si les règles d'authentification de canal en déterminent le contraire.
5. Si un module JAAS a été configuré, il est appelé
6. MQ Vérification des droits d'accès CONNECT appliquée à l'ID utilisateur MQ résultant.
7. Connexion établie avec une identité IBM MQ supposée.

Publication d'un message

Les étapes suivantes décrivent ce qui se passe lorsqu'un message est publié par un client AMQP. Les étapes déterminent si la connexion se poursuit et quelle identité IBM MQ l'application utilise pour les vérifications des droits d'accès:

1. Le cadre de connexion de lien AMQP arrive dans le gestionnaire de files d'attente. Le droit de publication IBM MQ pour la chaîne de rubrique spécifiée est vérifié pour l'identité de l'utilisateur MQ établie lors de la connexion.
2. Le message est publié dans la chaîne de rubrique spécifiée.

Abonnement à un modèle de rubrique

Les étapes suivantes décrivent ce qui se passe lorsqu'un client AMQP s'abonne à un modèle de rubrique. Les étapes déterminent si la connexion se poursuit et quelle identité IBM MQ l'application utilise pour les vérifications des droits d'accès:

1. Le cadre de connexion de lien AMQP arrive dans le gestionnaire de files d'attente. Le droit d'abonnement IBM MQ pour le modèle de rubrique spécifié est vérifié pour l'identité de l'utilisateur MQ établie lors de la connexion.
2. L'abonnement est créé.

MQ Light Identité et autorisation du client

Utilisez l'ID client MQ Light , le nom d'utilisateur MQ Light ou une identité client commune définie sur le canal ou dans une règle d'authentification de canal pour l'autorisation d'accès aux objets IBM MQ .

L'administrateur fait le choix lors de la définition ou de la modification du canal AMQP, en configurant le paramètre CONNAUTH du gestionnaire de files d'attente ou en définissant des règles d'authentification de canal. L'identité est utilisée pour autoriser l'accès aux rubriques IBM MQ. Le choix se fait en fonction des éléments suivants:

1. Attribut USECLNTID du canal.
2. Attribut ADOPTCTX de la règle CONNAUTH du gestionnaire de files d'attente.
3. Attribut MCAUSER défini sur le canal.
4. Attribut USERSRC d'une règle d'authentification de canal correspondante.

Eviter les incidents : L'identité choisie par ce processus est ensuite appelée, par exemple par la commande DISPLAY CHSTATUS (AMQP), MCAUSER du client. Sachez qu'il ne s'agit pas nécessairement de la même identité que le MCAUSER du canal auquel il est fait référence dans le choix (2).

Utilisez la commande IBM MQ **setmqaut** pour sélectionner les objets et les actions qui sont autorisés à être utilisés par l'identité associée au canal AMQP. Par exemple, les commandes suivantes autorisent une identité de canal AMQPClient, fournie par l'administrateur du gestionnaire de files d'attente QM1:

```
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p AMQPClient -all +pub +sub
```

et

```
setmqaut -m QM1 -t qmgr -p AMQPClient -all +connect
```

V 9.0.0 MQ Light Authentification du client à l'aide d'un mot de passe

Authentifiez le nom d'utilisateur MQ Light à l'aide du mot de passe du client. Vous pouvez authentifier le client à l'aide d'une identité différente de celle utilisée pour autoriser le client à publier des rubriques et à s'y abonner.

Le service AMQP peut utiliser MQ CONNAUTH ou JAAS pour authentifier le nom d'utilisateur du client. Si l'un d'eux est configuré, le mot de passe fourni par le client est vérifié par la configuration MQ CONNAUTH ou le module JAAS .

La procédure suivante décrit des exemples d'étapes permettant d'authentifier des utilisateurs individuels auprès des utilisateurs et des mots de passe du système d'exploitation local et, en cas de réussite, d'adopter l'identité commune AMQPUser:

1. L'administrateur IBM MQ définit l'identité MCAUSER du canal AMQP sur n'importe quel nom, tel que AMQPUser, à l'aide de IBM MQ Explorer.
2. L'administrateur IBM MQ autorise AMQPUser à publier et à s'abonner à n'importe quelle rubrique:

```
setmqaut -m QM1 -t topic -n SYSTEM.BASE.TOPIC -p AMQPUser -all +pub +sub +connect
```

3. L'administrateur IBM MQ configure une règle IDPWOS CONNAUTH pour vérifier le nom d'utilisateur et le mot de passe présentés par le client. La règle CONNAUTH doit définir CHCKCLNT (REQUIRED) et ADOPTCTX (NO).

Remarque : Il est recommandé d'utiliser des règles d'authentification de canal et de définir l'attribut de canal MCAUSER sur un utilisateur qui ne dispose pas de privilèges, afin d'autoriser un contrôle plus important sur les connexions au gestionnaire de files d'attente.

V 9.0.0 Confidentialité des publications sur les canaux

La confidentialité des publications AMQP envoyées dans les deux sens sur les canaux AMQP est sécurisée à l'aide de TLS pour chiffrer les transmissions sur la connexion.

Les clients AMQP qui se connectent aux canaux AMQP utilisent TLS pour sécuriser la confidentialité des publications transmises sur le canal à l'aide de la cryptographie à clé symétrique. Etant donné que les noeuds finaux ne sont pas authentifiés, vous ne pouvez pas accréditer le chiffrement de canal seul. Combinez la sécurisation de la confidentialité avec l'authentification du serveur ou l'authentification mutuelle.

Au lieu d'utiliser TLS, certains types de réseau privé virtuel (VPN), tels que IPsec, authentifient les noeuds finaux d'une connexion TCP/IP. Le réseau privé virtuel VPN chiffre chaque paquet transitant sur le réseau. Une fois qu'une telle connexion VPN est établie, vous avez établi un réseau sécurisé. Vous pouvez connecter des clients AMQP à des canaux AMQP à l'aide de TCP/IP sur le réseau VPN.

Le chiffrement des connexions TLS sans authentification du serveur expose la connexion aux attaques man-in-the-middle. Bien que les informations que vous échangez soient protégées contre les écoutes clandestines, vous ne savez pas avec qui vous réalisez les échanges. A moins que vous ne contrôliez le réseau, vous êtes exposé à ce que quelqu'un intercepte vos transmissions IP, en se faisant passer pour un noeud final.

Vous pouvez créer une connexion TLS chiffrée, sans authentifier le serveur, à l'aide d'un CipherSpec d'échange de clés Diffie-Hellman qui prend en charge le protocole TLS anonyme. Le secret maître, partagé entre le client et le serveur et utilisé pour chiffrer les transmissions TLS, est établi sans échanger de certificat de serveur signé de manière privée.

Etant donné que les connexions anonymes ne sont pas sécurisées, la plupart des implémentations TLS n'utilisent pas par défaut les CipherSpecs anonymes. Si une demande de client pour la connexion TLS est acceptée par un canal AMQP, le canal doit disposer d'un magasin de clés protégé par une phrase passe. Par défaut, les implémentations TLS n'utilisant pas de CipherSpecs anonymes, le magasin de clés doit contenir un certificat signé de manière privée que le client peut authentifier.

Si vous utilisez des spécifications CipherSpecs anonymes, le magasin de clés du serveur doit exister, mais il ne doit pas forcément contenir des certificats signés de manière privée.

Une autre façon d'établir une connexion cryptée consiste à remplacer le fournisseur d'accréditation côté client par votre propre implémentation. Votre fournisseur d'accréditation n'effectuerait pas l'authentification du certificat serveur, mais la connexion serait cryptée.

V 9.0.0 Configuration des clients MQ Light avec TLS

Vous pouvez configurer les clients MQ Light pour qu'ils utilisent TLS afin de protéger les données circulant sur le réseau et d'authentifier l'identité du gestionnaire de files d'attente auquel le client se connecte.

Pour utiliser TLS pour la connexion d'un client MQ Light à un canal AMQP, vous devez vous assurer que le gestionnaire de files d'attente a été configuré pour TLS. La rubrique [Configuration de TLS sur les gestionnaires de files d'attente](#) explique comment configurer le magasin de clés à partir duquel un gestionnaire de files d'attente lit les certificats TLS.

Lorsque le gestionnaire de files d'attente a été configuré avec un magasin de clés, vous devez configurer les attributs TLS sur le canal AMQP auquel les clients se connecteront. Les canaux AMQP ont quatre attributs liés à la configuration TLS comme suit:

SSLCAUTH

L'attribut SSLCAUTH permet de spécifier si le gestionnaire de files d'attente doit avoir besoin d'un client MQ Light pour présenter un certificat client afin de vérifier son identité.

SSLCIPH

L'attribut SSLCIPH spécifie le chiffrement que le canal doit utiliser pour coder les données dans le flux TLS.

SSLPEER

L'attribut SSLPEER est utilisé pour spécifier le nom distinctif (DN) auquel un certificat client doit correspondre si une connexion doit être autorisée.

CERTLABL

Le certificat CERTLABL indique le certificat que le gestionnaire de files d'attente doit présenter au client. Le magasin de clés du gestionnaire de files d'attente peut contenir plusieurs certificats. Cet attribut vous permet de spécifier le certificat à utiliser pour les connexions à ce canal. Si aucun CERTLABL n'est spécifié, le certificat du référentiel de clés du gestionnaire de files d'attente dont le libellé correspond à l'attribut CERTLABL du gestionnaire de files d'attente est utilisé.

Une fois que vous avez configuré votre canal AMQP avec les attributs TLS, vous devez redémarrer le service AMQP à l'aide de la commande suivante:

```
STOP SERVICE(SYSTEM.AMQP.SERVICE) START SERVICE(SYSTEM.AMQP.SERVICE)
```

Lorsqu'un client MQ Light se connecte à un canal AMQP protégé par TLS, le client vérifie l'identité du certificat présenté par le gestionnaire de files d'attente. Pour ce faire, vous devez configurer votre client MQ Light avec un magasin de clés de confiance contenant le certificat du gestionnaire de files d'attente. Les étapes à suivre varient en fonction du client MQ Light que vous utilisez.

- Pour la documentation de l'API MQ Light client for Node JS, voir <https://www.npmjs.com/package/mqlight>

- Pour la documentation de l'API MQ Light client for Java, voir <https://mqlight.github.io/java-mqlight/>
- Pour la documentation du client MQ Light pour Ruby, voir <https://www.rubydoc.info/github/mqlight/ruby-mqlight/>
- Pour la documentation MQ Light client for Python , voir <https://python-mqlight.readthedocs.org/en/latest/>

V 9.0.0 Déconnexion des clients MQ Light du gestionnaire de files d'attente

Si vous souhaitez déconnecter MQ Light du gestionnaire de files d'attente, exécutez la commande PURGE CHANNEL ou arrêtez la connexion au client MQ Light .

- Exécutez la commande **PURGE CHANNEL**. Exemple :

```
PURGE CHANNEL(MYAMQP) CLIENTID('recv_28dbb7e')
```

- Vous pouvez également arrêter la connexion que le client MQ Light utilise pour déconnecter le client en procédant comme suit:
 1. Recherchez la connexion utilisée par le client en exécutant la commande **DISPLAY CONN** . Exemple :

```
DISPLAY CONN(*) TYPE(CONN) WHERE (CLIENTID EQ 'recv_28dbb7e')
```

La sortie de la commande est la suivante:

```
DISPLAY CONN(*) TYPE(CONN) WHERE(CLIENTID EQ 'recv_28dbb7e')
40 : DISPLAY CONN(*) TYPE(CONN) WHERE(CLIENTID EQ 'recv_28dbb7e')
AMQ8276: Display Connection details.
CONN(707E0A565F2D0020)
EXTCONN(414D5143514D312020202020202020)
TYPE(CONN)
CLIENTID(recv_28dbb7e)
```

2. Arrêtez la connexion. Exemple :

```
STOP CONN(707E0A565F2D0020)
```

Administration de la multidiffusion

Utilisez ces informations pour en savoir plus sur les tâches d'administration de multidiffusion IBM MQ , telles que la réduction de la taille des messages de multidiffusion et l'activation de la conversion de données.

Initiation à la multidiffusion

Utilisez ces informations pour vous initier aux rubriques de multidiffusion IBM MQ et aux objets d'informations de communication.

Pourquoi et quand exécuter cette tâche

La messagerie IBM MQ Multicast utilise le réseau pour distribuer des messages en mappant des rubriques à des adresses de groupe. Les tâches suivantes permettent de tester rapidement si l'adresse IP et le port requis sont correctement configurés pour la messagerie multidiffusion.

Création d'un objet COMMINFO pour la multidiffusion

L'objet informations de communication (COMMINFO) contient les attributs associés à la transmission multidiffusion. Pour plus d'informations sur les paramètres de l'objet COMMINFO, voir [DEFINE COMMINFO](#).

Utilisez l'exemple de ligne de commande suivant pour définir un objet COMMINFO pour la multidiffusion:

```
DEFINE COMMINFO(MC1) GRPADDR(group address) PORT(port number)
```

où *MC1* est le nom de votre objet COMMINFO, *adresse de groupe* est l'adresse IP de multidiffusion de votre groupe ou le nom DNS, et le *numéro de port* est le port sur lequel effectuer la transmission (la valeur par défaut est 1414).

Un nouvel objet COMMINFO appelé *MC1* est créé. Il s'agit du nom que vous devez spécifier lors de la définition d'un objet TOPIC dans l'exemple suivant.

Création d'un objet TOPIC pour la multidiffusion

Une rubrique est l'objet des informations publiées dans un message de publication/abonnement, et une rubrique se définit en créant un objet TOPIC. Les objets TOPIC ont deux paramètres qui définissent s'ils peuvent être utilisés avec la multidiffusion ou non. Ces paramètres sont: **COMMINFO** et **MCAST**.

- **COMMINFO** Ce paramètre indique le nom de l'objet d'information de communication multidiffusion. Pour plus d'informations sur les paramètres de l'objet COMMINFO, voir [DEFINE COMMINFO](#).
- **MCAST** Ce paramètre indique si la multidiffusion est autorisée à ce niveau dans l'arborescence des rubriques.

Utilisez l'exemple de ligne de commande suivant pour définir un objet TOPIC pour la multidiffusion:

```
DEFINE TOPIC(ALLSPORTS) TOPICSTR('Sports') COMMINFO(MC1) MCAST(ENABLED)
```

Un nouvel objet TOPIC appelé *ALLSPORTS* est créé. Il comporte une chaîne de rubrique *Sports*, son objet d'informations de communication associé est appelé *MC1* (nom que vous avez spécifié lors de la définition d'un objet COMMINFO dans l'exemple précédent) et la multidiffusion est activée.

Test de la publication / abonnement multidiffusion

Une fois les objets TOPIC et COMMINFO créés, ils peuvent être testés à l'aide de l'exemple *amqspubc* et de l'exemple *amqssubc*. Pour plus d'informations sur ces exemples, voir [Exemples de programmes de publication / abonnement](#).

1. Ouvrez deux fenêtres de ligne de commande ; la première ligne de commande concerne l'exemple de publication *amqspubc* et la deuxième ligne de commande concerne l'exemple d'abonnement *amqssubc*.
2. Entrez la commande suivante sur la ligne de commande 1:

```
amqspubc Sports QM1
```

où *Sports* est la chaîne de rubrique de l'objet TOPIC défini dans un exemple précédent et *QM1* est le nom du gestionnaire de files d'attente.

3. Entrez la commande suivante sur la ligne de commande 2:

```
amqssubc Sports QM1
```

où *Sports* et *QM1* sont identiques à ceux utilisés à l'étape «2», à la page 247.

4. Entrez *Hello world* sur la ligne de commande 1. Si le port et l'adresse IP spécifiés dans l'objet COMMINFO sont configurés correctement, l'exemple *amqssubc*, qui écoute sur le port les publications à partir de l'adresse spécifiée, génère *Hello world* sur la ligne de commande 2.

Topologie de rubrique de multidiffusion IBM MQ

Utilisez cet exemple pour comprendre la topologie de rubrique IBM MQ Multicast.

La prise en charge de la multidiffusion IBM MQ requiert que chaque sous-arborescence ait son propre groupe de multidiffusion et son propre flux de données au sein de la hiérarchie totale.

Le schéma d'adressage IP *réseau avec classe* désigne un espace adresse pour l'adresse de multidiffusion. La plage de multidiffusion complète d'adresses IP est 224.0.0.0 à 239.255.255.255, mais certaines de ces adresses sont réservées. Pour la liste des adresses réservées, prenez contact avec votre administrateur système ou consultez la page <https://www.iana.org/assignments/multicast-addresses> pour plus d'informations. Il est recommandé d'utiliser l'adresse de multidiffusion locale dans la plage de 239.0.0.0 à 239.255.255.255.

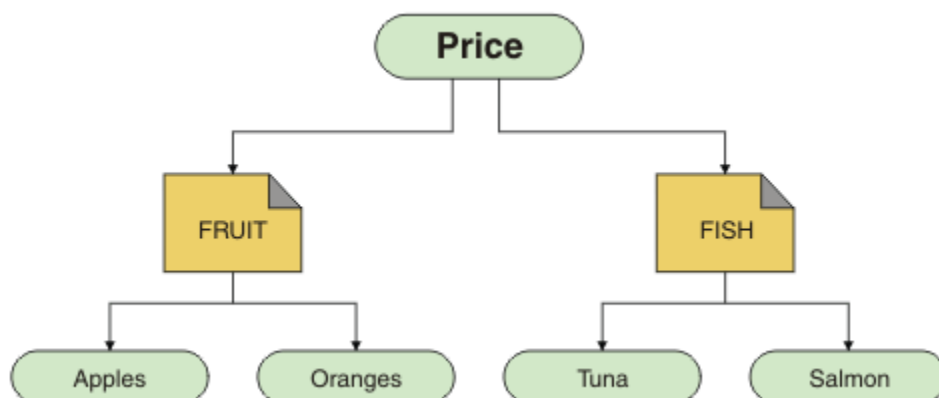
Dans le diagramme suivant, il existe deux flux de données de multidiffusion possibles:

```
DEF COMMINFO(MC1) GRPADDR(239.XXX.XXX.XXX
)
DEF COMMINFO(MC2) GRPADDR(239.YYY.YYY.YYY)
```

où 239.XXX.XXX.XXX et 239.YYY.YYY.YYY sont des adresses de multidiffusion valides.

Ces définitions de rubrique sont utilisées pour créer une arborescence de rubriques, comme illustré dans le diagramme suivant:

```
DEFINE TOPIC(FRUIT) TOPICSTRING('Price/FRUIT') MCAST(ENABLED) COMMINFO(MC1)
DEFINE TOPIC(FISH) TOPICSTRING('Price/FISH') MCAST(ENABLED) COMMINFO(MC2)
```



Chaque objet d'informations de communication multidiffusion (COMMINFO) représente un flux de données différent car leurs adresses de groupe sont différentes. Dans cet exemple, la rubrique FRUIT est définie pour utiliser l'objet COMMINFO MC1, la rubrique FISH est définie pour utiliser l'objet COMMINFO MC2 et le nœud Price n'a pas de définitions de multidiffusion.

IBM MQ La multidiffusion est limitée à 255 caractères pour les chaînes de rubrique. Cette limitation signifie que les noms des nœuds et des nœuds feuilles doivent être pris en compte dans l'arborescence. Si les noms des nœuds et des nœuds feuilles sont trop longs, la chaîne de rubrique peut dépasser 255 caractères et renvoyer le code anomalie 2425 (0979) (RC2425): MQRC_TOPIC_STRING_ERROR. Il est recommandé de rendre les chaînes de rubrique aussi courtes que possible car des chaînes de rubrique plus longues peuvent avoir un effet négatif sur les performances.

Contrôle de la taille des messages de multidiffusion

Utilisez ces informations pour en savoir plus sur le format de message IBM MQ et réduire la taille des messages IBM MQ.

Les messages IBM MQ sont associés à un certain nombre d'attributs qui sont contenus dans le descripteur de message. Pour les messages de petite taille, ces attributs peuvent représenter la majeure partie du trafic de données et peuvent avoir un effet négatif significatif sur le débit de transmission. IBM

MQ Multicast permet à l'utilisateur de configurer les attributs qui, le cas échéant, sont transmis avec le message.

La présence d'attributs de message, autres que la chaîne de rubrique, varie selon que l'objet COMMINFO indique qu'ils doivent être envoyés ou non. Si un attribut n'est pas transmis, l'application de réception applique une valeur par défaut. Les valeurs MQMD par défaut ne sont pas nécessairement identiques à la valeur MQMD_DEFAULT et sont décrites plus loin dans [Tableau 17](#), à la page 249.

L'objet COMMINFO contient l'attribut MCPROP qui contrôle le nombre de zones MQMD et de propriétés utilisateur qui circulent avec le message. En définissant la valeur de cet attribut sur un niveau approprié, vous pouvez contrôler la taille des messages de multidiffusion IBM MQ :

MCPROP

Les propriétés de multidiffusion contrôlent la quantité de propriétés MQMD et propriétés utilisateur émises avec le message.

TOUT

Toutes les propriétés utilisateur et toutes les zones du MQMD sont transmises.

REPONSE

Seules les propriétés utilisateur et les zones MQMD liées à la réponse des messages sont transmises. Ces propriétés sont :

- MsgType
- MessageId
- CorrelId
- ReplyToQ
- ReplyToQmgr

UTILISATEUR

Seules les propriétés utilisateur sont transmises.

AUCUN

Aucune propriété utilisateur ni zone MQMD n'est transmise.

COMPAT

Cette valeur entraîne la transmission du message dans un mode compatible à RMM, ce qui permet une certaine interopérabilité avec les applications XMS et IBM Integration Bus RMM en cours.

Attributs de message de multidiffusion

Les attributs de message peuvent provenir de différents emplacements, tels que MQMD, les zones de MQRFH2 et les propriétés de message.

Le tableau suivant montre ce qui se passe lorsque des messages sont envoyés en fonction de la valeur de MCPROP et de la valeur par défaut utilisée lorsqu'un attribut n'est pas envoyé.

| Attribut | Action lors de l'utilisation de la multidiffusion | Valeur par défaut si non transmise |
|-----------------|--|---|
| TopicString | Toujours inclus | Non applicable |
| MQMQ StrucId | Non transmis | Non applicable |
| Version MQMD | Non transmis | Non applicable |
| Rapport | Inclus si non par défaut | 0 |
| MsgType | Inclus si non par défaut | MQMT_DATAGRAM |
| Expiration | Inclus si non par défaut | 0 |
| Commentaires | Inclus si non par défaut | 0 |

Tableau 17. Attributs de messagerie et comment ils sont liés à la multidiffusion (suite)

| Attribut | Action lors de l'utilisation de la multidiffusion | Valeur par défaut si non transmise |
|-----------------|---|------------------------------------|
| Codage | Inclus si non par défaut | MQENC_NORMAL (équivalent) |
| CodedCharSetId | Inclus si non par défaut | 1208 |
| Format | Inclus si non par défaut | MQRFH2 |
| Priorité | Inclus si non par défaut | 4 |
| Persistance | Inclus si non par défaut | MQPER_NON_PERSISTENT |
| MsgId | Inclus si non par défaut | Null |
| CorrelId | Inclus si non par défaut | Null |
| BackoutCount | Inclus si non par défaut | 0 |
| ReplyToQ | Inclus si non par défaut | Blanc |
| ReplyToQMgr | Inclus si non par défaut | Blanc |
| UserIdentifier | Inclus si non par défaut | Blanc |
| AccountingToken | Inclus si non par défaut | Null |
| PutAppIType | Inclus si non par défaut | MQAT_JAVA |
| PutAppIName | Inclus si non par défaut | Blanc |
| PutDate | Inclus si non par défaut | Blanc |
| PutTime | Inclus si non par défaut | Blanc |
| ApplOriginData | Inclus si non par défaut | Blanc |
| GroupID | Exclu | Non applicable |
| MsgSeqNumber | Exclu | Non applicable |
| Décalage | Exclu | Non applicable |
| MsgFlags | Exclu | Non applicable |
| OriginalLength | Exclu | Non applicable |
| UserProperties | Inclus | Non applicable |

Référence associée

[ALTER COMMINFO](#)

Information associée

[DEFINI COMMINFO](#)

Activation de la conversion de données pour la messagerie multidiffusion

Utilisez ces informations pour comprendre le fonctionnement de la conversion de données pour la messagerie IBM MQ Multicast.

IBM MQ Multicast est un protocole partagé et sans connexion. Par conséquent, il n'est pas possible pour chaque client d'effectuer des demandes spécifiques de conversion de données. Chaque client abonné au même flux de multidiffusion reçoit les mêmes données binaires ; par conséquent, si une conversion de données IBM MQ est requise, la conversion est effectuée localement sur chaque client.

Dans une installation à plateforme mixte, il se peut que la plupart des clients requièrent les données dans un format qui n'est pas le format natif de l'application de transmission. Dans cette situation, les valeurs

CCSID et **ENCODING** de l'objet **COMMINFO** de multidiffusion peuvent être utilisées pour définir le codage de la transmission de message à des fins d'efficacité.

IBM MQ Multicast prend en charge la conversion de données de la charge de message pour les formats intégrés suivants:

- MQADMIN
- MQEVENT
- MQPCF
- MQRFH
- MQRFH2
- MQSTR

En plus de ces formats, vous pouvez également définir vos propres formats et utiliser un exit de conversion de données [MQDXP-Paramètre d'exit de conversion de données](#) .

Pour plus d'informations sur la programmation des conversions de données, voir [Conversion de données dans l'interface MQI pour la messagerie multidiffusion](#).

Pour plus d'informations sur la conversion de données, voir [Conversion de données](#).

Pour plus d'informations sur les exits de conversion de données et `ClientExitPath`, voir la section [ClientExitPath](#) du fichier de configuration client.

Surveillance des applications de multidiffusion

Utilisez ces informations pour en savoir plus sur l'administration et la surveillance de la multidiffusion IBM MQ .

L'état des diffuseurs actuels et des abonnés pour le trafic multidiffusion (par exemple le nombre de messages envoyés et reçus, ou le nombre de messages perdus) est périodiquement transmis au serveur depuis le client. Lorsque le statut est reçu, l'attribut `COMMEV` de l'objet **COMMINFO** indique si le gestionnaire de files d'attente insère ou non un message d'événement dans `SYSTEM.ADMIN.PUBSUB.EVENT`. Le message d'événement contient les informations de statut reçues. Ces informations sont un outil de diagnostic précieux pour trouver la source d'un problème.

Utilisez la commande MQSC **DISPLAY CONN** pour afficher les informations de connexion relatives aux applications connectées au gestionnaire de files d'attente. Pour plus d'informations sur la commande **DISPLAY CONN** , voir [DISPLAY CONN](#).

Utilisez la commande MQSC **DISPLAY TPSTATUS** pour afficher le statut de vos diffuseurs de publications et de vos abonnés. Pour plus d'informations sur la commande **DISPLAY TPSTATUS** , voir [DISPLAY TPSTATUS](#).

COMMEV et l'indicateur de fiabilité des messages multidiffusion

L' *indicateur de fiabilité*, utilisé conjointement avec l'attribut **COMMEV** de l'objet **COMMINFO**, est un élément clé de la surveillance des diffuseurs de publications et des abonnés IBM MQ Multicast. L'indicateur de fiabilité (zone **MSGREL** renvoyée dans les commandes de statut de publication ou d'abonnement) est un indicateur IBM MQ qui illustre le pourcentage de transmissions qui ne comportent pas d'erreurs. Parfois, les messages doivent être retransmis en raison d'une erreur de transmission, ce qui se reflète dans la valeur de **MSGREL**. Les causes potentielles des erreurs de transmission incluent les abonnés lents, les réseaux occupés et les indisponibilités du réseau. **COMMEV** contrôle si les messages d'événement sont générés pour les descripteurs de multidiffusion créés à l'aide de l'objet **COMMINFO** et sont définis sur l'une des trois valeurs possibles:

DESACTIVE

Les messages d'événement ne sont pas écrits.

Activée

Les messages d'événement sont toujours écrits, avec une fréquence définie dans le paramètre **COMMINFO MONINT** .

EXCEPTION

Les messages d'événement sont écrits si la fiabilité des messages est inférieure au seuil de fiabilité. Un niveau de fiabilité de message inférieur ou égal à 90% indique qu'il peut y avoir un problème avec la configuration réseau ou qu'une ou plusieurs applications de publication / abonnement s'exécutent trop lentement:

- La valeur **MSGREL (100, 100)** indique qu'il n'y a eu aucun problème à court terme ou à long terme.
- La valeur **MSGREL (80, 60)** indique que 20% des messages présentent actuellement des problèmes, mais qu'il s'agit également d'une amélioration par rapport à la valeur à long terme de 60.

Les clients peuvent continuer à transmettre et à recevoir du trafic multidiffusion même lorsque la connexion monodiffusion au gestionnaire de files d'attente est interrompue. Par conséquent, les données peuvent être obsolètes.

Fiabilité des messages de multidiffusion

Utilisez ces informations pour apprendre à définir l'abonnement IBM MQ Multicast et l'historique des messages.

La mise en mémoire tampon des données transmises (historique des messages à conserver à l'extrémité émettrice de la liaison) par IBM MQ est un élément clé pour surmonter l'échec de la transmission avec la multidiffusion. Ce processus signifie qu'aucune mise en mémoire tampon des messages n'est requise dans le processus d'application d'insertion car IBM MQ fournit la fiabilité. La taille de cet historique est configurée via l'objet d'informations de communication (COMMINFO), comme décrit dans les informations suivantes. Un tampon de transmission plus grand signifie qu'il y a plus d'historique de transmission à retransmettre si nécessaire, mais en raison de la nature de la multidiffusion, une distribution assurée à 100% ne peut pas être prise en charge.

L'historique des messages de multidiffusion IBM MQ est contrôlé dans l'objet d'informations de communication (COMMINFO) par l'attribut **MSGHIST** :

MSGHIST

Cette valeur correspond à la quantité d'historique des messages, en kilooctets, qui est conservée par le système pour gérer les retransmissions dans le cas des accusés de réception négatifs.

La valeur 0 indique le niveau de fiabilité le moins élevé. La valeur par défaut est 100 Ko.

L'historique des nouveaux abonnements IBM MQ Multicast est contrôlé dans l'objet d'informations de communication (COMMINFO) par l'attribut **NSUBHIST** :

NSUBHIST

L'historique du nouvel abonné indique si un abonné rejoignant un flot de publication reçoit autant de données que disponible, ou reçoit uniquement les publications effectuées à partir du moment de l'abonnement.

AUCUN

La valeur NONE permet à l'émetteur de transmettre uniquement la publication effectuée à partir du moment de l'abonnement. NONE est la valeur utilisée par défaut.

TOUT

La valeur ALL permet à l'émetteur de retransmettre autant d'historique de la rubrique que connu. Dans certaines circonstances, cette situation peut donner un comportement similaire à celui des publications conservées.

Remarque : L'utilisation de la valeur de ALL peut avoir un impact négatif sur les performances s'il existe un historique de sujet volumineux car tout l'historique de sujet est retransmis.

Information associée

[DEFINI COMMINFO](#)

[ALTER COMMINFO](#)

Tâches de multidiffusion avancées

Utilisez ces informations pour en savoir plus sur les tâches d'administration IBM MQ Multicast avancées telles que la configuration des fichiers `.ini` et l'interopérabilité avec IBM MQ LLM.

Pour plus d'informations sur la sécurité dans une installation de multidiffusion, voir [Sécurité de multidiffusion](#).

Pontage entre les domaines de publication / abonnement multidiffusion et non multidiffusion

Utilisez ces informations pour comprendre ce qui se passe lorsqu'un diffuseur de publications non multidiffusion est publié dans une rubrique activée pour la multidiffusion IBM MQ .

Si un diffuseur de publications non multidiffusion est publié dans une rubrique définie comme **MCAST** activée et **BRIDGE** activée, le gestionnaire de files d'attente transmet le message en mode multidiffusion directement aux abonnés qui peuvent être en mode écoute. Un diffuseur de publications multidiffusion ne peut pas publier dans des rubriques qui ne sont pas activées pour la multidiffusion.

Les rubriques existantes peuvent être activées en mode multidiffusion en définissant les paramètres **MCAST** et **COMMINFO** d'un objet de rubrique. Pour plus d'informations sur ces paramètres, voir [Concepts de multidiffusion initiaux](#) .

L'attribut **BRIDGE** de l'objet **COMMINFO** contrôle les publications des applications qui n'utilisent pas la multidiffusion. Si **BRIDGE** est défini sur **ENABLED** et que le paramètre **MCAST** de la rubrique est également défini sur **ENABLED**, les publications des applications qui n'utilisent pas la multidiffusion sont routées vers les applications qui le font. Pour plus d'informations sur le paramètre **BRIDGE** , voir [DEFINE COMMINFO](#).

Configuration des fichiers .ini pour la multidiffusion

Utilisez ces informations pour comprendre les zones de multidiffusion IBM MQ dans les fichiers `.ini` .

Une configuration IBM MQ Multicast supplémentaire peut être effectuée dans un fichier `ini` . Le fichier `ini` spécifique que vous devez utiliser dépend du type d'application:

- Client: configurez le fichier `MQ_DATA_PATH/mqclient.ini` .
- Gestionnaire de files d'attente: configurez le fichier `MQ_DATA_PATH/qmgrs/QMNAME/qm.ini` .

où `MQ_DATA_PATH` est l'emplacement du répertoire de données IBM MQ (`/var/mqm/mqclient.ini`) et `QMNAME` est le nom du gestionnaire de files d'attente auquel s'applique le fichier `.ini` .

Le fichier `.ini` contient des zones utilisées pour affiner le comportement de IBM MQ Multicast:

```
Multicast:
Protocol      = IP | UDP
IPVersion     = IPv4 | IPv6 | ANY | BOTH
LimitTransRate = DISABLED | STATIC | DYNAMIC
TransRateLimit = 100000
SocketTTL     = 1
Batch         = NO
Loop          = 1
Interface     = <IPAddress>
FeedbackMode  = ACK | NACK | WAIT1
HeartbeatTimeout = 20000
HeartbeatInterval = 2000
```

Protocole

UDP

Dans ce mode, les paquets sont envoyés à l'aide du protocole UDP. Les éléments de réseau ne peuvent pas fournir d'assistance dans la distribution multidiffusion comme ils le font en mode IP. Le format de paquet reste compatible avec PGM. Il s'agit de la valeur par défaut.

Adresse IP

Dans ce mode, l'émetteur envoie des paquets IP bruts. Les éléments de réseau avec prise en charge de PGM aident à la distribution de paquets multidiffusion fiable. Ce mode est entièrement compatible avec la norme PGM.

IPVersion**IPv4**

Communiquez uniquement à l'aide du protocole IPv4 . Il s'agit de la valeur par défaut.

IPv6

Communiquez uniquement à l'aide du protocole IPv6 .

ANY

Communiquez à l'aide de IPv4, de IPv6 ou des deux, selon le protocole disponible.

LES DEUX

Prend en charge la communication à l'aide de IPv4 et de IPv6.

LimitTransDébit**DESACTIVE**

Il n'y a pas de contrôle du débit de transmission. Il s'agit de la valeur par défaut.

STATIQUE

Implémente le contrôle du débit de transmission statique. L'émetteur ne transmet pas à un débit supérieur au débit spécifié par le paramètre de limite TransRate.

DYNAMIQUE

L'émetteur adapte son débit de transmission en fonction de la rétroaction qu'il reçoit des récepteurs. Dans ce cas, la limite de débit de transmission ne peut pas être supérieure à la valeur spécifiée par le paramètre TransRateLimit. L'émetteur tente d'atteindre un débit de transmission optimal.

Limite TransRate

Limite de débit de transmission en kbit/s.

SocketTTL

La valeur de SocketTTL détermine si le trafic de multidiffusion peut passer par un routeur, ou le nombre de routeurs qu'il peut traverser.

Lot

Contrôle si les messages sont envoyés par lots ou immédiatement. Il existe 2 valeurs possibles:

- *NON* Les messages ne sont pas envoyés par lots, ils sont envoyés immédiatement.
- *OUI* Les messages sont traités par lots.

Boucle

Définissez la valeur sur 1 pour activer la boucle de multidiffusion. La boucle de multidiffusion définit si les données envoyées sont ou non rebouclées sur l'hôte.

utilisateur

Adresse IP de l'interface sur laquelle transite le trafic multidiffusion. Pour plus d'informations et pour résoudre les problèmes, voir: [Test d'applications multidiffusion sur un réseau non multidiffusion et Définition du réseau approprié pour le trafic multidiffusion](#)

FeedbackMode**NACK**

Commentaires en retour par des accusés de réception négatifs. Il s'agit de la valeur par défaut.

Accusé de réception

Commentaires en retour par des accusés de réception positifs.

WAIT1

Rétroaction par des accusés de réception positifs où l'émetteur attend seulement 1 ACK de n'importe lequel des récepteurs.

HeartbeatTimeout

Délai d'attente du signal de présence en millisecondes. La valeur 0 indique que les événements de délai de signal de présence ne sont pas émis par le ou les récepteurs de la rubrique. La valeur par défaut est 20000.

HeartbeatInterval

Intervalle des pulsations en millisecondes. La valeur 0 indique qu'aucun signal de présence n'est envoyé. L'intervalle des pulsations doit être considérablement inférieur à la valeur **HeartbeatTimeout** pour éviter les faux événements de délai d'attente des pulsations. La valeur par défaut est 2000.

Interopérabilité multidiffusion avec la messagerie à faible temps d'attente IBM MQ

Utilisez ces informations pour comprendre l'interopérabilité entre IBM MQ Multicast et IBM MQ Low Latency Messaging (LLM).

Le transfert de contenu de base est possible pour une application utilisant LLM, avec une autre application utilisant la multidiffusion pour échanger des messages dans les deux sens. Bien que la multidiffusion utilise la technologie LLM, le produit LLM lui-même n'est pas intégré. Par conséquent, il est possible d'installer LLM et IBM MQ Multicast, et d'utiliser et de traiter les deux produits séparément.

Les applications LLM qui communiquent avec la multidiffusion peuvent avoir besoin d'envoyer et de recevoir des propriétés de message. Les propriétés de message IBM MQ et les zones MQMD sont transmises en tant que propriétés de message LLM avec des codes de propriété de message LLM spécifiques, comme indiqué dans le tableau suivant:

| Propriété IBM MQ | IBM MQ Type de propriété LLM | Type de propriété LLM | Code de propriété LLM |
|---------------------|------------------------------|-------------------------|-----------------------|
| MQMD.Report | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1001 |
| MQMD.MsgType | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1002 |
| MQMD.Expiry | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1003 |
| MQMD.Feedback | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1004 |
| MQMD.Encoding | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1005 |
| MQMD.CodedCharSetId | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1006 |
| MQMD.Format | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_Chaîne | -1007 |
| MQMD.Priority | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1008 |
| MQMD.Persistence | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1009 |
| MQMD.MsgId | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_ByteArray | -1010 |
| MQMD.BackoutCount | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_Int32 | -1012 |
| MQMD.ReplyToQ | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_Chaîne | -1013 |
| MQMD.ReplyToQMger | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_Chaîne | -1014 |
| MQMD.PutDate | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_Chaîne | -1020 |
| MQMD.PutTime | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_Chaîne | -1021 |
| MQMD.ApplOriginData | RMM_MSG_PROP_BYTES | LLM_PROP_KIND_Chaîne | -1022 |
| MQPubOptions | RMM_MSG_PROP_INT32 | LLM_PROP_KIND_int32 | -1053 |

Pour plus d'informations sur LLM, voir la documentation du produit LLM: [IBM MQ Low Latency Messaging](#).

Présente les méthodes disponibles pour administrer IBM MQ sur IBM i.

Les tâches d'administration incluent la création, le démarrage, la modification, l'affichage, l'arrêt et la suppression de clusters, de processus et d'objets IBM MQ (gestionnaires de files d'attente, files d'attente, listes de noms, définitions de processus, canaux, canaux de connexion client, programmes d'écoute, services et objets d'informations d'authentification).

Pour plus de détails sur l'administration de IBM MQ for IBM i, voir les liens suivants:

- [«Gestion d'IBM MQ for IBM i à l'aide de commandes CL»](#), à la page 256
- [«Autres méthodes d'administration de IBM MQ for IBM i»](#), à la page 270
- [«Gestion des travaux pour IBM i»](#), à la page 276

Concepts associés

[«Disponibilité, sauvegarde, récupération et redémarrage sur IBM i»](#), à la page 284

Utilisez ces informations pour comprendre comment IBM MQ for IBM i utilise le support de journalisation IBM i pour aider sa stratégie de sauvegarde et de restauration.

Référence associée

[«mise au repos IBM MQ for IBM i»](#), à la page 330

Cette section explique comment mettre au repos (arrêt approprié) IBM MQ for IBM i.

Information associée

[Modification des informations de configuration sous IBM i](#)

[Description des noms de bibliothèque de gestionnaire de files d'attente IBM MQ for IBM i](#)

[Configuration de la sécurité sous IBM i](#)

[Gestionnaire de files d'attente de messages non livrés sur IBM i](#)

[Identification des problèmes liés aux applications IBM MQ for IBM i](#)

[Services et composants installables sous IBM i](#)

[Objets système et par défaut sous IBM i](#)

Gestion d'IBM MQ for IBM i à l'aide de commandes CL

Utilisez ces informations pour comprendre les commandes IBM MQ IBM i .

La plupart des groupes de commandes IBM MQ , y compris ceux associés aux gestionnaires de files d'attente, aux files d'attente, aux rubriques, aux canaux, aux listes de noms, aux définitions de processus et aux objets d'informations d'authentification, sont accessibles à l'aide de la commande **WRK*** appropriée.

La commande principale de l'ensemble est **WRKMQM**. Cette commande permet, par exemple, d'afficher la liste de tous les gestionnaires de files d'attente du système, ainsi que des informations de statut. Vous pouvez également traiter toutes les commandes spécifiques au gestionnaire de files d'attente à l'aide de différentes options pour chaque entrée.

A partir de la commande **WRKMQM** , vous pouvez sélectionner des zones spécifiques de chaque gestionnaire de files d'attente, par exemple, en utilisant des canaux, des rubriques ou des files d'attente, et à partir de là, sélectionner des objets individuels.

Enregistrement des définitions d'application IBM MQ

Lorsque vous créez ou personnalisez des applications IBM MQ , il est utile de conserver un enregistrement de toutes les définitions IBM MQ créées. Cet enregistrement peut être utilisé pour:

- Objectifs de reprise
- Mise à jour
- Déploiement d'applications IBM MQ

Vous pouvez enregistrer les définitions d'application IBM MQ de l'une des deux manières suivantes:

1. Création de programmes CL pour générer vos définitions IBM MQ pour le serveur.
2. Création de fichiers texte MQSC en tant que membres SRC pour générer vos définitions IBM MQ à l'aide du langage de commande IBM MQ multiplateforme.

Pour plus de détails sur la définition des objets de file d'attente, voir [«Commandes de script \(MQSC\)»](#), à la page 11 et [«Utilisation des formats de commande programmables IBM MQ»](#), à la page 23.

Information associée

[Référence des commandes CL IBM MQ for IBM i](#)

Avant de commencer à utiliser IBM MQ for IBM i à l'aide de commandes CL

Utilisez ces informations pour démarrer le sous-système IBM MQ et créer un gestionnaire de files d'attente local.

Avant de commencer

Vérifiez que le sous-système IBM MQ est en cours d'exécution (à l'aide de la commande STRSBS QMQM/QMQM) et que la file d'attente de travaux associée à ce sous-système n'est pas suspendue. Par défaut, le sous-système IBM MQ et la file d'attente de travaux sont tous deux nommés QMQM dans la bibliothèque QMQM.

Pourquoi et quand exécuter cette tâche

Utilisation de la ligne de commande IBM i pour démarrer un gestionnaire de files d'attente

Procédure

1. Créez un gestionnaire de files d'attente local en émettant la commande CRTMQM à partir d'une ligne de commande IBM i.
Lorsque vous créez un gestionnaire de files d'attente, vous avez la possibilité de faire de ce gestionnaire de files d'attente le gestionnaire de files d'attente par défaut. Le gestionnaire de files d'attente par défaut (dont un seul peut exister) est le gestionnaire de files d'attente auquel s'applique une commande CL, si le paramètre de nom de gestionnaire de files d'attente (MQMNAME) est omis.
2. Démarrez un gestionnaire de files d'attente local en exécutant la commande STRMQM à partir d'une ligne de commande IBM i.
Si le démarrage du gestionnaire de files d'attente prend plus de quelques secondes, IBM MQ affiche des messages d'état détaillant par intermittence la progression du démarrage. Pour plus d'informations sur ces messages, voir [Messages et codes anomalie](#).

Que faire ensuite

Vous pouvez arrêter un gestionnaire de files d'attente en émettant la commande ENDMQM à partir de la ligne de commande IBM i et en contrôlant un gestionnaire de files d'attente en émettant d'autres commandes IBM MQ à partir d'une ligne de commande IBM i.

Les gestionnaires de files d'attente éloignées ne peuvent pas être démarrés à distance mais doivent être créés et démarrés sur leurs systèmes par des opérateurs locaux. Une exception à cette règle est l'existence de fonctions d'exploitation à distance (en dehors de IBM MQ for IBM i) permettant d'activer de telles opérations.

L'administrateur de file d'attente locale ne peut pas arrêter un gestionnaire de files d'attente éloignées.

Remarque : Dans le cadre de la mise au repos d'un système IBM MQ, vous devez mettre au repos les gestionnaires de files d'attente actifs. Ceci est décrit dans [«mise au repos IBM MQ for IBM i»](#), à la page 330.

Utilisez ces informations pour comprendre les méthodes de création d'objets IBM MQ pour IBM i.

Avant de commencer

Les tâches suivantes suggèrent différentes manières d'utiliser IBM MQ for IBM i à partir de la ligne de commande.

Pourquoi et quand exécuter cette tâche

Il existe deux méthodes en ligne pour créer des objets IBM MQ , à savoir:

Procédure

1. A l'aide d'une commande Create, par exemple: la commande **Create MQM Queue : CRTMQMQ**
2. A l'aide d'une commande Work with MQM object, suivie de F6, par exemple: La commande **Work with MQM Queues : WRKMQM**

Que faire ensuite

Pour obtenir la liste de toutes les commandes, voir [Commandes CL IBM MQ for IBM i](#).

Remarque : Toutes les commandes MQM peuvent être soumises à partir du menu Commandes gestionnaire de files d'attente de messages. Pour afficher ce menu, entrez GO CMDMQM sur la ligne de commande et appuyez sur la touche Enter .

Le système affiche automatiquement le panneau d'invite lorsque vous sélectionnez une commande dans ce menu. Pour afficher le panneau d'invite d'une commande que vous avez saisie directement sur la ligne de commande, appuyez sur F4 avant d'appuyer sur la touche Enter .

Création d'une file d'attente locale à l'aide de la commande CRTMQMQ

Procédure

1. Entrez CHGMQM sur la ligne de commande et appuyez sur la touche F4 .
2. Dans le panneau **Créer une file d'attente MQM**, entrez le nom de la file d'attente à créer dans la zone Queue name . Pour indiquer un nom à casse mixte, placez le nom entre apostrophes.
3. Entrez *LCL dans la zone Queue type .
4. Indiquez un nom de gestionnaire de files d'attente, sauf si vous utilisez le gestionnaire de files d'attente par défaut, puis appuyez sur la touche Enter . Vous pouvez remplacer n'importe quelle valeur par une nouvelle valeur. Faites défiler vers l'avant pour afficher d'autres zones. Les options utilisées pour les clusters se trouvent à la fin de la liste des options.
5. Lorsque vous avez modifié des valeurs, appuyez sur la touche Enter pour créer la file d'attente.

Création d'une file d'attente locale à l'aide de la commande WRKMQM

Procédure

1. Entrez WRKMQM sur la ligne de commande.
2. Entrez le nom d'un gestionnaire de files d'attente.
3. Si vous souhaitez afficher le panneau d'invite, appuyez sur F4. Le panneau d'invite permet de réduire le nombre de files d'attente affichées en indiquant un nom de file d'attente générique ou un type de file d'attente.
4. Appuyez sur Enter et le panneau **Work with MQM Queues** s'affiche. Vous pouvez remplacer n'importe quelle valeur par une nouvelle valeur. Faites défiler vers l'avant pour afficher d'autres zones. Les options utilisées pour les clusters se trouvent à la fin de la liste des options.

5. Appuyez sur F6 pour créer une nouvelle file d'attente. Vous accédez alors au panneau **CRTMQMQ**. Pour savoir comment créer la file d'attente, voir «Création d'une file d'attente locale à l'aide de la commande CRTMQMQ», à la page 258. Une fois que vous avez créé la file d'attente, le **panneau Work with MQM Queues** s'affiche à nouveau. La nouvelle file d'attente est ajoutée à la liste lorsque vous appuyez sur F5=Refresh.

Modification des attributs du gestionnaire de files d'attente

Pourquoi et quand exécuter cette tâche

Pour modifier les attributs du gestionnaire de files d'attente spécifié dans la commande **CHGMQM**, indiquez les attributs et les valeurs à modifier. Par exemple, utilisez les options suivantes pour modifier les attributs de `jupiter.queue.manager`:

Procédure

Entrez **CHGMQM** sur la ligne de commande et appuyez sur la touche F4.

Résultats

La commande modifie la file d'attente de rebut utilisée et active les événements d'interdiction.

Utilisation des files d'attente locales sous IBM i

Cette section contient des exemples de certaines des commandes que vous pouvez utiliser pour gérer les files d'attente locales. Toutes les commandes affichées sont également disponibles à l'aide des options du **panneau de commande WRKMQMQ**.

Définition d'une file d'attente locale

Pour une application, le gestionnaire de files d'attente local est le gestionnaire de files d'attente auquel l'application est connectée. Les files d'attente gérées par le gestionnaire de files d'attente local sont dites locales pour ce gestionnaire de files d'attente.

Utilisez la commande **CRTMQMQ QTYPE *LCL** pour créer une définition de file d'attente locale et pour créer la structure de données appelée file d'attente. Vous pouvez également modifier les caractéristiques de la file d'attente à partir de celles de la file d'attente locale par défaut.

Dans cet exemple, la file d'attente que nous définissons, `orange.local.queue`, possède les caractéristiques suivantes:

- Il est activé pour les extractions, désactivé pour les insertions et fonctionne sur la base du premier entré, premier sorti (FIFO).
- Il s'agit d'une file d'attente *ordinaire*, c'est-à-dire qu'il ne s'agit pas d'une file d'attente d'initialisation ou de transmission et qu'elle ne génère pas de messages de déclenchement.
- La longueur maximale de la file d'attente est de 1000 messages ; la longueur maximale est de 2000 octets.

La commande suivante effectue cette opération sur le gestionnaire de files d'attente par défaut:

```
CRTMQMQ QNAME('orange.local.queue') QTYPE(*LCL)
TEXT('Queue for messages from other systems')
PUTENBL(*NO)
GETENBL(*YES)
TRGENBL(*NO)
MSGDLYSEQ(*FIFO)
MAXDEPTH(1000)
MAXMSGLEN(2000)
USAGE(*NORMAL)
```

Remarque :

1. USAGE *NORMAL indique que cette file d'attente n'est pas une file d'attente de transmission.

2. Si vous disposez déjà d'une file d'attente locale portant le nom `orange.local.queue` sur le même gestionnaire de files d'attente, cette commande échoue. Utilisez l'attribut `REPLACE *YES` si vous souhaitez remplacer la définition existante d'une file d'attente, mais voir aussi [«Modification des attributs de file d'attente locale»](#), à la page 261.

Définition d'une file d'attente de rebut

Chaque gestionnaire de files d'attente doit disposer d'une file d'attente locale à utiliser comme file d'attente de rebut pour que les messages qui ne peuvent pas être distribués à leur destination correcte puissent être stockés en vue d'une extraction ultérieure. Vous devez indiquer explicitement au gestionnaire de files d'attente la file d'attente de rebut. Pour ce faire, vous pouvez spécifier une file d'attente de rebut dans la commande **CRTMQM** ou utiliser la commande **CHGMQM** pour en spécifier une ultérieurement. Vous devez également définir la file d'attente de rebut avant de pouvoir l'utiliser.

Un exemple de file d'attente de rebut appelé `SYSTEM.DEAD.LETTER.QUEUE` est fourni avec le produit. Cette file d'attente est automatiquement créée lorsque vous créez le gestionnaire de files d'attente. Vous pouvez modifier cette définition si nécessaire. Il n'est pas nécessaire de le renommer, même si vous pouvez le faire si vous le souhaitez.

Une file d'attente de rebut n'a pas d'exigences particulières si ce n'est que:

- Il doit s'agir d'une file d'attente locale.
- Son attribut `MAXMSGL` (longueur maximale de message) doit permettre à la file d'attente de prendre en charge les messages les plus volumineux que le gestionnaire de files d'attente doit traiter **plus** la taille de l'en-tête de rebut (`MQDLH`).

IBM MQ fournit un gestionnaire de files d'attente de rebut qui permet d'indiquer comment les messages trouvés dans une file d'attente de rebut doivent être traités ou supprimés. Pour plus d'informations, voir [Le gestionnaire de files d'attente de rebut IBM MQ for IBM i](#).

Affichage des attributs d'objet par défaut

Lorsque vous définissez un objet IBM MQ, il prend tous les attributs que vous ne spécifiez pas à partir de l'objet par défaut. Par exemple, lorsque vous définissez une file d'attente locale, celle-ci hérite de tous les attributs que vous omettez dans la définition de la file d'attente locale par défaut, appelée `SYSTEM.DEFAULT.LOCAL.QUEUE`. Pour voir exactement quels sont ces attributs, utilisez la commande suivante:

```
DSPMQM QNAME(SYSTEM.DEFAULT.LOCAL.QUEUE) MQMNAME(MYQUEUEMANAGER)
```

Copie d'une définition de file d'attente locale

Vous pouvez copier une définition de file d'attente à l'aide de la commande `CPYMQM`. Exemple :

```
CPYMQM FROMQ('orange.local.queue') TOQ('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Cette commande crée une file d'attente avec les mêmes attributs que notre file d'attente d'origine `orange.local.queue`, plutôt que ceux de la file d'attente locale par défaut du système.

Vous pouvez également utiliser la commande **CPYMQM** pour copier une définition de file d'attente, mais en remplaçant une ou plusieurs modifications apportées aux attributs de l'original. Exemple :

```
CPYMQM FROMQ('orange.local.queue') TOQ('third.queue') MQMNAME(MYQUEUEMANAGER)  
MAXMSGLEN(1024)
```

Cette commande copie les attributs de la file d'attente `orange.local.queue` dans la file d'attente `third.queue`, mais indique que la longueur maximale des messages dans la nouvelle file d'attente doit être de 1024 octets au lieu de 2000.

Remarque : Lorsque vous utilisez la commande **CPYMQMQ**, vous copiez uniquement les attributs de file d'attente et non les messages de la file d'attente.

Modification des attributs de file d'attente locale

Vous pouvez modifier les attributs de file d'attente de deux manières, à l'aide de la commande **CHGMQMQ** ou de la commande **CPYMQMQ** avec l'attribut REPLACE *YES. Dans «[Définition d'une file d'attente locale](#)», à la page 259, vous avez défini la file d'attente orange.local.queue. Si, par exemple, vous devez augmenter la longueur maximale des messages dans cette file d'attente à 10 000 octets.

- A l'aide de la commande **CHGMQMQ** :

```
CHGMQMQ QNAME('orange.local.queue') MQMNAME(MYQUEUEMANAGER) MAXMSGLEN(10000)
```

Cette commande modifie un attribut unique, celui de la longueur maximale de message ; tous les autres attributs restent identiques.

- A l'aide de la commande **CRTMQMQ** avec l'option REPLACE *YES, par exemple:

```
CRTMQMQ QNAME('orange.local.queue') QTYPE(*LCL) MQMNAME(MYQUEUEMANAGER)  
MAXMSGLEN(10000) REPLACE(*YES)
```

Cette commande modifie non seulement la longueur maximale des messages, mais également tous les autres attributs, auxquels sont attribuées leurs valeurs par défaut. La file d'attente est maintenant mise en file d'attente activée alors qu'auparavant elle était mise en file d'attente interdite. L'insertion activée est la valeur par défaut, comme indiqué par la file d'attente SYSTEM.DEFAULT.LOCAL.QUEUE, sauf si vous l'avez modifiée.

Si vous **réduisez** la longueur maximale des messages dans une file d'attente existante, les messages existants ne sont pas affectés. Toutefois, tout nouveau message doit répondre aux nouveaux critères.

Effacement d'une file d'attente locale

Pour supprimer tous les messages d'une file d'attente locale appelée magenta.queue, utilisez la commande suivante:

```
CLRMQMQ QNAME('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Vous ne pouvez pas effacer une file d'attente si:

- Des messages non validés ont été insérés dans la file d'attente lors d'un point de synchronisation.
- La file d'attente est ouverte par une application.

Suppression d'une file d'attente locale

Utilisez la commande **DLTMQMQ** pour supprimer une file d'attente locale.

Une file d'attente ne peut pas être supprimée si elle contient des messages non validés ou si elle est en cours d'utilisation.

Activation des files d'attente volumineuses

IBM MQ prend en charge les files d'attente supérieures à 2 Go. Consultez la documentation de votre système d'exploitation pour savoir comment activer IBM i pour la prise en charge des fichiers volumineux.

La documentation du produit IBM i est disponible dans [IBM Documentation](#).

Certains utilitaires peuvent ne pas être en mesure de gérer les fichiers de plus de 2 Go. Avant d'activer la prise en charge des fichiers volumineux, consultez la documentation de votre système d'exploitation pour plus d'informations sur les restrictions liées à cette prise en charge.

Cette section contient des exemples de certaines des commandes que vous pouvez utiliser pour gérer les files d'attente alias. Toutes les commandes affichées sont également disponibles à l'aide des options du **panneau de commande WRKMQM**.

Une file d'attente alias (parfois appelée alias de file d'attente) fournit une méthode de redirection des appels MQI. Une file d'attente alias n'est pas une file d'attente réelle, mais une définition qui se résout en file d'attente réelle. La définition de file d'attente alias contient un nom de file d'attente cible, qui est spécifié par l'attribut TGTQNAME.

Lorsqu'une application spécifie une file d'attente alias dans un appel MQI, le gestionnaire de files d'attente résout le nom de la file d'attente réelle lors de l'exécution.

Par exemple, une application a été développée pour placer des messages dans une file d'attente appelée `my.alias.queue`. Il indique le nom de cette file d'attente lorsqu'il émet une demande **MQOPEN** et, indirectement, s'il insère un message dans cette file d'attente. L'application ne sait pas que la file d'attente est une file d'attente d'alias. Pour chaque appel MQI utilisant cet alias, le gestionnaire de files d'attente résout le nom de la file d'attente réelle, qui peut être une file d'attente locale ou une file d'attente éloignée définie sur ce gestionnaire de files d'attente.

En modifiant la valeur de l'attribut TGTQNAME, vous pouvez rediriger les appels MQI vers une autre file d'attente, éventuellement sur un autre gestionnaire de files d'attente. Cela est utile pour la maintenance, la migration et l'équilibrage de charge.

Définition d'une file d'attente alias

La commande suivante crée une file d'attente alias:

```
CRTMQMQ QNAME('my.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
MQMNAME(MYQUEUEMANAGER)
```

Cette commande redirige les appels MQI qui spécifient `my.alias.queue` vers la file d'attente `yellow.queue`. La commande ne crée pas la file d'attente cible ; les appels MQI échouent si la file d'attente `yellow.queue` n'existe pas lors de l'exécution.

Si vous modifiez la définition d'alias, vous pouvez rediriger les appels MQI vers une autre file d'attente. Exemple :

```
CHGMQM QNAME('my.alias.queue') TGTQNAME('magenta.queue') MQMNAME(MYQUEUEMANAGER)
```

Cette commande redirige les appels MQI vers une autre file d'attente, `magenta.queue`.

Vous pouvez également utiliser des files d'attente alias pour qu'une seule file d'attente (la file d'attente cible) ait des attributs différents pour différentes applications. Pour ce faire, vous définissez deux alias, un pour chaque application. Supposons qu'il existe deux applications:

- L'application ALPHA peut insérer des messages dans `yellow.queue`, mais elle n'est pas autorisée à en extraire des messages.
- L'application BETA peut extraire des messages de `yellow.queue`, mais elle n'est pas autorisée à y placer des messages.

Pour ce faire, utilisez les commandes suivantes:

```
/* This alias is put enabled and get disabled for application ALPHA */
CRTMQMQ QNAME('alphas.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
PUTENBL(*YES) GETENBL(*NO) MQMNAME(MYQUEUEMANAGER)

/* This alias is put disabled and get enabled for application BETA */
```

```
CRTMQMQ QNAME('betas.alias.queue') QTYPE(*ALS) TGTQNAME('yellow.queue')
PUTENBL(*NO) GETENBL(*YES) MQMNAME(MYQUEUEMANAGER)
```

ALPHA utilise le nom de file d'attente `alphas.alias.queue` dans ses appels MQI ; BETA utilise le nom de file d'attente `betas.alias.queue`. Ils accèdent tous deux à la même file d'attente, mais de différentes manières.

Vous pouvez utiliser l'attribut `REPLACE *YES` lorsque vous définissez des files d'attente alias, de la même manière que vous utilisez ces attributs avec des files d'attente locales.

Utilisation d'autres commandes avec des files d'attente alias

Vous pouvez utiliser les commandes appropriées pour afficher ou modifier les attributs de file d'attente alias. Exemple :

```
* Display the alias queue's attributes */
DSPMQMQ QNAME('alphas.alias.queue') MQMNAME(MYQUEUEMANAGER)
/* ALTER the base queue name, to which the alias resolves. */
/* FORCE = Force the change even if the queue is open. */
CHQMCMQ QNAME('alphas.alias.queue') TGTQNAME('orange.local.queue') FORCE(*YES)
MQMNAME(MYQUEUEMANAGER)
```

IBM i

Utilisation des files d'attente modèles sous IBM i

Cette section contient des exemples de certaines des commandes que vous pouvez utiliser pour gérer les files d'attente modèles. Toutes les commandes affichées sont également disponibles à l'aide des options du **panneau de commande WRKMCMQ**.

Un gestionnaire de files d'attente crée une file d'attente dynamique s'il reçoit un appel MQI d'une application spécifiant un nom de file d'attente qui a été défini comme file d'attente modèle. Le nom de la nouvelle file d'attente dynamique est généré par le gestionnaire de files d'attente lors de la création de la file d'attente. Une file d'attente modèle est un modèle qui spécifie les attributs des files d'attente dynamiques créées à partir de celle-ci.

Les files d'attente modèles constituent une méthode pratique permettant aux applications de créer des files d'attente au fur et à mesure qu'elles sont requises.

Définition d'une file d'attente modèle

Vous définissez une file d'attente modèle avec un ensemble d'attributs de la même manière que vous définissez une file d'attente locale. Les files d'attente modèles et les files d'attente locales ont le même ensemble d'attributs, sauf que dans les files d'attente modèles, vous pouvez indiquer si les files d'attente dynamiques créées sont temporaires ou permanentes. (Les files d'attente permanentes sont gérées lors des redémarrages du gestionnaire de files d'attente, alors que les files d'attente temporaires ne le sont pas). Exemple :

```
CRTMQMQ QNAME('green.model.queue') QTYPE(*MDL) DFNTYPE(*PERMDYN)
```

Cette commande crée une définition de file d'attente modèle. A partir de l'attribut `DFNTYPE`, les files d'attente réelles créées à partir de ce modèle sont des files d'attente dynamiques permanentes. Les attributs non spécifiés sont automatiquement copiés à partir de la file d'attente par défaut `SYSYSTEM.DEFAULT.MODEL.QUEUE`.

Vous pouvez utiliser l'attribut `REPLACE *YES` lorsque vous définissez des files d'attente modèles, de la même manière que vous les utilisez avec des files d'attente locales.

Utilisation d'autres commandes avec des files d'attente modèles

Vous pouvez utiliser les commandes appropriées pour afficher ou modifier les attributs d'une file d'attente modèle. Exemple :

```
/* Display the model queue's attributes */
DSPMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('green.model.queue')

/* ALTER the model queue to enable puts on any */
/* dynamic queue created from this model. */
CHGMQM MQMNAME(MYQUEUEMANAGER) QNAME('blue.model.queue') PUTENBL(*YES)
```

IBM i

Utilisation du déclenchement sous IBM i

Utilisez ces informations pour en savoir plus sur le déclenchement et les définitions de processus.

IBM MQ fournit une fonction permettant de démarrer automatiquement une application lorsque certaines conditions d'une file d'attente sont remplies. Par exemple, lorsque le nombre de messages dans une file d'attente atteint un nombre spécifié. Cette fonction est appelée *déclenchement* et est décrite en détail dans [Déclenchement des canaux](#).

Qu'est-ce qui se déclenche?

Le gestionnaire de files d'attente définit certaines conditions comme constituant des événements déclencheurs. Si le déclenchement est activé pour une file d'attente et qu'un événement déclencheur se produit, le gestionnaire de files d'attente envoie un message de déclenchement à une file d'attente appelée file d'attente d'initialisation. La présence du message de déclenchement dans la file d'attente d'initialisation indique qu'un événement déclencheur s'est produit.

Les messages de déclenchement générés par le gestionnaire de files d'attente ne sont pas persistants. Cela a pour effet de réduire la consignation (améliorant ainsi les performances) et de réduire les doublons lors du redémarrage, ce qui améliore le temps de redémarrage.

Qu'est-ce que le moniteur de déclenchement?

Le programme qui traite la file d'attente d'initialisation est appelé une application de moniteur de déclenchement et sa fonction est de lire le message de déclenchement et de prendre les mesures appropriées, en fonction des informations contenues dans le message de déclenchement. Normalement, cette action consiste à démarrer une autre application pour traiter la file d'attente à l'origine de la génération du message de déclenchement. Du point de vue du gestionnaire de files d'attente, il n'y a rien de spécial à propos de l'application du moniteur de déclenchement: c'est une autre application qui lit les messages d'une file d'attente (la file d'attente d'initialisation).

Modification des attributs de soumission de travaux du moniteur de déclenchement

Le moniteur de déclenchement fourni en tant que commande **STRMQMTRM** soumet un travail pour chaque message de déclenchement à l'aide de la description de travail par défaut du système, QDFTJOB. Cela comporte des limitations dans la mesure où les travaux soumis sont toujours appelés QDFTJOB et possèdent les attributs de la description de travail par défaut, y compris la liste des bibliothèques, *SYSVAL. IBM MQ fournit une méthode pour remplacer ces attributs. Par exemple, il est possible de personnaliser les travaux soumis pour qu'ils aient des noms de travail plus significatifs, comme suit:

1. Dans la description de travail, indiquez la description de votre choix, par exemple les valeurs de consignation.
2. Indiquez les données d'environnement de la définition de processus utilisée dans le processus de déclenchement:

```
CHGMQMPRC PRCNAME(MY_PROCESS) MQMNAME(MHA3) ENVDATA ('JOB(MYLIB/TRIGJOB)')
```


Le moniteur de déclenchement exécute un SBMJOB à l'aide de la description spécifiée.

Il est possible de remplacer d'autres attributs du SBMJOB en spécifiant le mot clé et la valeur appropriés dans les données d'environnement de la définition de processus. La seule exception à cette règle est le mot clé CMD car cet attribut est rempli par le moniteur de déclenchement. Voici un exemple de commande permettant de spécifier les données d'environnement de la définition de processus dans laquelle le nom et la description du travail doivent être modifiés:

```
CHGMQMPRC PRCNAME(MY_PROCESS) MQMNAME(MHA3) ENVDATA ('JOB(MYLIB/TRIGJOB)
JOB(TRIGGER)')
```

Définition d'une file d'attente d'application pour le déclenchement

Une file d'attente d'application est une file d'attente locale utilisée par les applications pour la messagerie, via l'interface MQI. Le déclenchement requiert la définition d'un certain nombre d'attributs de file d'attente dans la file d'attente d'application. Le déclenchement est activé par l'attribut TRGENBL .

Dans cet exemple, un événement déclencheur doit être généré lorsqu'il y a 100 messages de priorité 5 ou supérieure dans la file d'attente locale `motor.insurance.queue`, comme suit:

```
CRTMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('motor.insurance.queue') QTYPE(*LCL)
PRCNAME('motor.insurance.quote.process') MAXMSGLEN(2000)
DFTMSGPST(*YES) INITQNAME('motor.ins.init.queue')
TRGENBL(*YES) TRGTYPE(*DEPTH) TRGDEPTH(100) TRGMSGPTY(5)
```

où les paramètres sont :

MQMNAME (MYQUEUEMANAGER)

Nom du gestionnaire de files d'attente.

QNAME ('motor.insurance.queue')

Nom de la file d'attente d'application en cours de définition.

PRCNAME ('motor.insurance.quote.process')

Nom de l'application devant être démarrée par un programme de moniteur de déclenchement.

MAXMSGLEN (2000)

Longueur maximale des messages dans la file d'attente.

DFTMSGPST (*YES)

Les messages de cette file d'attente sont persistants par défaut.

INITQNAME ('motor.ins.init.queue')

Nom de la file d'attente d'initialisation dans laquelle le gestionnaire de files d'attente doit insérer le message de déclenchement.

TRGENBL (*YES)

Valeur de l'attribut de déclencheur.

TRGTYPE (*DEPTH)

Un événement déclencheur est généré lorsque le nombre de messages de la priorité requise (**TRGMSGPTY**) atteint le nombre spécifié dans **TRGDEPTH**.

TRGDEPTH (100)

Nombre de messages requis pour générer un événement déclencheur.

TRGMSGPTY (5)

Priorité des messages qui doivent être comptés par le gestionnaire de files d'attente pour déterminer s'il convient de générer un événement déclencheur. Seuls les messages de priorité 5 ou supérieure sont comptés.

Définition d'une file d'attente d'initialisation

Lorsqu'un événement déclencheur se produit, le gestionnaire de files d'attente place un message de déclenchement dans la file d'attente d'initialisation spécifiée dans la définition de file d'attente

d'application. Les files d'attente d'initialisation n'ont pas de paramètres spéciaux, mais vous pouvez utiliser la définition suivante de la file d'attente locale `motor.ins.init.queue` pour obtenir de l'aide:

```
CRTMQMQ MQMNAME(MYQUEUEMANAGER) QNAME('motor.ins.init.queue') QTYPE(*LCL)
GETENBL(*YES) SHARE(*NO) TRGTYPE(*NONE)
MAXMSGL(2000)
MAXDEPTH(1000)
```

Création d'une définition de processus

La commande **CRTMQMPC** permet de créer une définition de processus. Une définition de processus associe une file d'attente d'application à l'application qui doit traiter les messages de la file d'attente. Cette opération est effectuée via l'attribut **PRCNAME** de la file d'attente d'application `motor.insurance.quote`. La commande suivante crée le processus requis, `motor.insurance.quote.process`, identifié dans cet exemple:

```
CRTMQMPC MQMNAME(MYQUEUEMANAGER) PRCNAME('motor.insurance.quote.process')
TEXT('Insurance request message processing')
APPTYPE(*OS400) APPID(MQTEST/TESTPROG)
USRDATA('open, close, 235')
```

où les paramètres sont :

MQMNAME(MYQUEUEMANAGER)

Nom du gestionnaire de files d'attente.

PRCNAME('motor.insurance.quote.process')

Nom de la définition de processus.

TEXT('Insurance request message processing')

Description du programme d'application auquel cette définition est liée. Ce texte s'affiche lorsque vous utilisez la commande **DSPMQMPC**. Cela peut vous aider à identifier ce que le processus fait. Si vous utilisez des espaces dans la chaîne, vous devez placer la chaîne entre apostrophes.

APPTYPE(*OS400)

Type d'application à démarrer.

APPID(MQTEST/TESTPROG)

Nom du fichier exécutable de l'application, spécifié en tant que nom de fichier qualifié complet.

USRDATA('open, close, 235')

Données définies par l'utilisateur, qui peuvent être utilisées par l'application.

Affichage de votre définition de processus

Utilisez la commande **DSPMQMPC** pour examiner les résultats de votre définition. Exemple :

```
MQMNAME(MYQUEUEMANAGER) DSPMQMPC('motor.insurance.quote.process')
```

Vous pouvez également utiliser la commande **CHGMQMPC** pour modifier une définition de processus existante et la commande **DLTMQMPC** pour supprimer une définition de processus.



Communication entre deux systèmes IBM MQ sous IBM i

Cet exemple de codage montre comment configurer deux systèmes IBM MQ for IBM i, à l'aide de commandes CL, pour qu'ils puissent communiquer entre eux.

Les systèmes sont appelés SYSTEMA et SYSTEMB et le protocole de communication utilisé est TCP/IP.

Effectuez la procédure suivante :

1. Créez un gestionnaire de files d'attente sous SYSTEMA, en l'appelant QMGRA1.

```
CRTMQM  MQMNAME(QMGRA1) TEXT('System A - Queue +
Manager 1') UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
```

2. Démarrez ce gestionnaire de files d'attente.

```
STRMQM  MQMNAME(QMGRA1)
```

3. Définissez les objets IBM MQ sur SYSTEMA dont vous avez besoin pour envoyer des messages à un gestionnaire de files d'attente sur SYSTEMB.

```
/* Transmission queue */
CRTMQMQ  QNAME(XMITQ.TO.QMGRB1) QTYPE(*LCL) +
MQMNAME(QMGRA1) TEXT('Transmission Queue +
to QMGRB1') MAXDEPTH(5000) USAGE(*TMQ)

/* Remote queue that points to a queue called TARGETB */
/* TARGETB belongs to queue manager QMGRB1 on SYSTEMB */
CRTMQMQ  QNAME(TARGETB.ON.QMGRB1) QTYPE(*RMT) +
MQMNAME(QMGRA1) TEXT('Remote Q pointing +
at Q TARGETB on QMGRB1 on Remote System +
SYSTEMB') RMTQNAME(TARGETB) +
RMTMQMNAME(QMGRB1) TMQNAME(XMITQ.TO.QMGRB1)

/* TCP/IP sender channel to send messages to the queue manager on SYSTEMB*/
CRTMQMCHL CHLNAME(QMGRA1.TO.QMGRB1) CHLTYPE(*SDR) +
MQMNAME(QMGRA1) TRPTYPE(*TCP) +
TEXT('Sender Channel From QMGRA1 on +
SYSTEMA to QMGRB1 on SYSTEMB') +
CONNNAME(SYSTEMB) TMQNAME(XMITQ.TO.QMGRB1)
```

4. Créez un gestionnaire de files d'attente sous SYSTEMB, en l'appelant QMGRB1.

```
CRTMQM  MQMNAME(QMGRB1) TEXT('System B - Queue +
Manager 1') UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
```

5. Démarrez le gestionnaire de files d'attente sous SYSTEMB.

```
STRMQM  MQMNAME(QMGRB1)
```

6. Définissez les objets IBM MQ dont vous avez besoin pour recevoir des messages du gestionnaire de files d'attente sous SYSTEMA.

```
/* Local queue to receive messages on */
CRTMQMQ  QNAME(TARGETB) QTYPE(*LCL) MQMNAME(QMGRB1) +
TEXT('Sample Local Queue for QMGRB1')

/* Receiver channel of the same name as the sender channel on SYSTEMA */
CRTMQMCHL CHLNAME(QMGRA1.TO.QMGRB1) CHLTYPE(*RCVR) +
MQMNAME(QMGRB1) TRPTYPE(*TCP) +
TEXT('Receiver Channel from QMGRA1 to +
QMGRB1')
```

7. Enfin, démarrez un programme d'écoute TCP/IP sur SYSTEMB afin que le canal puisse être démarré. Cet exemple utilise le port par défaut 1414.

```
STRMQMLSR MQMNAME(QMGRB1)
```

Vous êtes maintenant prêt à envoyer des messages de test entre SYSTEMA et SYSTEMB. A l'aide de l'un des exemples fournis, placez une série de messages dans votre file d'attente éloignée sur SYSTEMA.

Démarrez le canal sur SYSTEMA, soit à l'aide de la commande **STRMQMCHL**, soit à l'aide de la commande **WRKMQMCHL** et en entrant une demande de démarrage (option 14) sur le canal émetteur.

Le canal doit passer à l'état RUNNING et les messages sont envoyés à la file d'attente TARGETB sur SYSTEMB.

Vérifiez vos messages en exécutant la commande suivante:

```
WRKMQMSG QNAME(TARGETB) MQMNAME(QMGRB1).
```

IBM i Exemples de définitions de ressource sous IBM i

Cet exemple contient l'exemple de programme CL AMQSAMP4 IBM i.

```
/* **** */
/*
/* Program name: AMQSAMP4
/*
/* Description: Sample CL program defining MQM queues
/* to use with the sample programs
/* Can be run, with changes as needed, after
/* starting the MQM
/*
/* <N_OCO_COPYRIGHT>
/* Licensed Materials - Property of IBM
/*
/* 63H9336
/* (c) Copyright IBM Corp. 1993, 2023. All Rights Reserved.
/*
/* US Government Users Restricted Rights - Use, duplication or
/* disclosure restricted by GSA ADP Schedule Contract with
/* IBM Corp.
/* <NOC_COPYRIGHT>
/*
/* **** */
/*
/* Function:
/*
/* AMQSAMP4 is a sample CL program to create or reset the
/* MQI resources to use with the sample programs.
/*
/* This program, or a similar one, can be run when the MQM
/* is started - it creates the objects if missing, or resets
/* their attributes to the prescribed values.
/*
/*
/* Exceptions signaled: none
/* Exceptions monitored: none
/*
/* AMQSAMP4 takes a single parameter, the Queue Manager name
/*
/* **** */
QSYS/PGM PARM(&QMGRNAME)

/* **** */
/* Queue Manager Name Parameter
/* **** */
QSYS/DCL VAR(&QMGRNAME) TYPE(*CHAR)

/* **** */
/* EXAMPLES OF DIFFERENT QUEUE TYPES
/*
/* Create local, alias and remote queues
/*
/* Uses system defaults for most attributes
/*
/* **** */
/* Create a local queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.LOCAL') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
+
TEXT('Sample local queue') /* description */+
SHARE(*YES) /* Shareable */+
DFTMSGPST(*YES) /* Persistent messages OK */
```

```

/* Create an alias queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.ALIAS')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*ALS) REPLACE(*YES)                 +
+
TEXT('Sample alias queue')                 +
DFTMSGPST(*YES) /* Persistent messages OK */+
TGTONAME('SYSTEM.SAMPLE.LOCAL')

/* Create a remote queue - in this case, an indirect reference */
/* is made to the sample local queue on OTHER queue manager */
CRTMQMQ QNAME('SYSTEM.SAMPLE.REMOTE')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*RMT) REPLACE(*YES)                 +
+
TEXT('Sample remote queue')/* description */+
DFTMSGPST(*YES) /* Persistent messages OK */+
RMTQNAME('SYSTEM.SAMPLE.LOCAL')           +
RMTMQMNAME(OTHER) /* Queue is on OTHER */

/* Create a transmission queue for messages to queues at OTHER */
/* By default, use remote node name */
CRTMQMQ QNAME('OTHER') /* transmission queue name */+
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES) +
TEXT('Transmission queue to OTHER') +
USAGE(*TMQ) /* transmission queue */

/*****
/* SPECIFIC QUEUES AND PROCESS USED BY SAMPLE PROGRAMS */
/* Create local queues used by sample programs */
/* Create MQI process associated with sample initiation queue */
*****/
/* General reply queue */
CRTMQMQ QNAME('SYSTEM.SAMPLE.REPLY')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('General reply queue')                 +
DFTMSGPST(*NO) /* Not Persistent */

/* Queue used by AMQSINQ4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.INQ')        +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('Queue for AMQSINQ4')                 +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO) /* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.INQPROCESS')       +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Queue used by AMQSSET4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.SET')        +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('Queue for AMQSSET4')                 +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO)/* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+
PRCNAME('SYSTEM.SAMPLE.SETPROCESS')       +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Queue used by AMQSECH4 */
CRTMQMQ QNAME('SYSTEM.SAMPLE.ECHO')      +
MQMNAME(&QMGRNAME)                        +
QTYPE(*LCL) REPLACE(*YES)                 +
+
TEXT('Queue for AMQSECH4')                 +
SHARE(*YES) /* Shareable */+
DFTMSGPST(*NO)/* Not Persistent */+
+
TRGENBL(*YES) /* Trigger control on */+
TRGTYPE(*FIRST)/* Trigger on first message*/+

```

```

PRCNAME('SYSTEM.SAMPLE.ECHOPROCESS') +
INITQNAME('SYSTEM.SAMPLE.TRIGGER')

/* Initiation Queue used by AMQSTRG4, sample trigger process */
CRTMQMQ QNAME('SYSTEM.SAMPLE.TRIGGER') +
MQMNAME(&QMGRNAME) +
QTYPE(*LCL) REPLACE(*YES) +
TEXT('Trigger queue for sample programs')

/* MQI Processes associated with triggered sample programs */
/*
/***** Note - there are versions of the triggered samples *****/
/***** in different languages - set APPID for these *****/
/***** process to the variation you want to trigger *****/
/*
CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.INQPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
TEXT('Trigger process for AMQSINQ4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSINQ4') /* C */ +
/* APPID('QMOM/AMQOINQ4') /* COBOL */ +
/* APPID('QMOM/AMQ3INQ4') /* RPG - ILE */

CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.SETPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
TEXT('Trigger process for AMQSSET4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSSET4') /* C */ +
/* APPID('QMOM/AMQOSET4') /* COBOL */ +
/* APPID('QMOM/AMQ3SET4') /* RPG - ILE */

CRTMQMPRC PRCNAME('SYSTEM.SAMPLE.ECHOPROCESS') +
MQMNAME(&QMGRNAME) +
REPLACE(*YES) +
TEXT('Trigger process for AMQSECH4') +
ENVDATA('JOBPTY(3)') /* Submit parameter */ +
/** Select the triggered program here **/ +
APPID('QMOM/AMQSECH4') /* C */ +
/* APPID('QMOM/AMQOECH4') /* COBOL */ +
/* APPID('QMOM/AMQ3ECH4') /* RPG - ILE */

/*****
/*
/* Normal return.
/*
/*****
SNDPGMMSG MSG('AMQSAMP4 Completed creating sample +
objects for ' *CAT &QMGRNAME)
RETURN
ENDPGM

/*****
/*
/* END OF AMQSAMP4
/*
/*****

```

IBM i Autres méthodes d'administration de IBM MQ for IBM i

L'utilisation de commandes CL est la méthode recommandée pour l'administration de IBM MQ for IBM i. Toutefois, vous pouvez utiliser diverses autres méthodes d'administration, y compris les commandes MQSC, les commandes PCF et l'administration à distance.

Vous utilisez normalement les commandes CL IBM i pour administrer IBM MQ for IBM i. Pour une présentation de ces commandes, voir [«Gestion d'IBM MQ for IBM i à l'aide de commandes CL»](#), à la page 256.

Vous pouvez utiliser les événements d'instrumentation IBM MQ pour surveiller le fonctionnement des gestionnaires de files d'attente. Pour plus d'informations sur les événements d'instrumentation IBM MQ et leur utilisation, voir [Événements d'instrumentation](#).

Vous pouvez utiliser l'une des méthodes d'administration décrites dans les sous-rubriques suivantes comme alternative à l'utilisation des commandes CL IBM i :

IBM i Administration locale et distante sous IBM i

Vous administrez les objets IBM MQ for IBM i en local ou à distance.

Administration locale signifie l'exécution de tâches d'administration sur tous les gestionnaires de files d'attente que vous avez définis sur votre système local. Dans IBM MQ, vous pouvez considérer cela comme une administration locale car aucun canal IBM MQ n'est impliqué, c'est-à-dire que la communication est gérée par le système d'exploitation. Pour effectuer ce type de tâche, vous devez soit vous connecter au système distant et émettre les commandes à partir de là, soit créer un processus qui peut émettre les commandes pour vous.

IBM MQ prend en charge l'administration à partir d'un point unique via ce qu'on appelle l' *administration à distance*. L'administration à distance consiste à envoyer des messages de contrôle PCF (Programmable Command Format) à SYSTEM.ADMIN.COMMAND.QUEUE sur le gestionnaire de files d'attente cible.

Il existe plusieurs manières de générer des messages PCF. Il s'agit des fonctions suivantes :

1. Ecriture d'un programme à l'aide de messages PCF. Voir [«Administration à l'aide de commandes PCF sous IBM i»](#), à la page 273.
2. Ecriture d'un programme à l'aide de MQAI, qui envoie des messages PCF. Voir [«Utilisation de MQAI pour simplifier l'utilisation des fichiers PCF»](#), à la page 35.
3. A l'aide de l'explorateur IBM MQ, disponible avec IBM MQ for Windows, vous pouvez utiliser une interface graphique et générer les messages PCF appropriés. Voir [«Utilisation du IBM MQ Explorer avec IBM MQ for IBM i»](#), à la page 273.
4. Utilisez **STRMQMQSC** pour envoyer des commandes indirectement à un gestionnaire de files d'attente éloignées. Voir [«Administration à l'aide de commandes MQSC sous IBM i»](#), à la page 271.

Par exemple, vous pouvez émettre une commande distante pour modifier une définition de file d'attente sur un gestionnaire de files d'attente éloignées.

Certaines commandes ne peuvent pas être émises de cette manière, notamment la création ou le démarrage de gestionnaires de files d'attente et de serveurs de commandes. Pour effectuer ce type de tâche, vous devez vous connecter au système distant et émettre les commandes à partir de celui-ci ou créer un processus qui peut émettre les commandes pour vous.

IBM i Administration à l'aide de commandes MQSC sous IBM i

Utilisez ces informations pour en savoir plus sur les commandes MQSC et pour savoir comment les utiliser pour administrer IBM MQ for IBM i.

Les commandes de script IBM MQ (MQSC) sont écrites dans un format lisible par l'utilisateur, c'est-à-dire dans du texte EBCDIC. Les commandes MQSC permettent de gérer les objets de gestionnaire de files d'attente, y compris le gestionnaire de files d'attente lui-même, les files d'attente, les définitions de processus, les listes de noms, les canaux, les canaux de connexion client, les programmes d'écoute, les services, les rubriques et les objets d'informations d'authentification.

Vous émettez des commandes MQSC vers un gestionnaire de files d'attente à l'aide de la commande CL **STRMQMQSC** IBM MQ. Cette méthode est une méthode de traitement par lots uniquement. Son entrée provient d'un fichier physique source du système de bibliothèque du serveur. Le nom par défaut de ce fichier physique source est QMQSC.



Avertissement : N'utilisez pas la bibliothèque QTEMP comme bibliothèque source pour STRMQMQSC, car l'utilisation de la bibliothèque QTEMP est limitée. Vous devez utiliser une autre bibliothèque comme fichier d'entrée de la commande.

IBM MQ for IBM i ne fournit pas de fichier source appelé QMQSC. Pour traiter les commandes MQSC, vous devez créer le fichier source QMQSC dans la bibliothèque de votre choix, en exécutant la commande suivante:

```
CRTSRCPF FILE(MYLIB/QMQSC) RCDLEN(240) TEXT('IBM MQ - MQSC Source')
```

La source MQSC est conservée dans les membres de ce fichier source. Pour utiliser les membres, entrez la commande suivante:

```
WRKMBRPDM MYLIB/QMQSC
```

Vous pouvez désormais ajouter de nouveaux membres et gérer les membres existants

Vous pouvez également entrer des commandes MQSC de manière interactive en exécutant RUNMQSC ou:

1. Entrez le nom du gestionnaire de files d'attente et appuyez sur la touche Enter pour accéder au panneau des résultats **WRKMQM**.
2. Sélection de F23=More options sur ce panneau.
3. Sélection de l'option 26 pour un gestionnaire de files d'attente actif dans le panneau présenté dans la Figure 32, à la page 272.

Pour mettre fin à une session MQSC de ce type, entrez end.

Figure 32, à la page 272 est un extrait d'un fichier de commandes MQSC affichant une commande MQSC (DEFINE QLOCAL) avec ses attributs.

```
.  
.   
DEFINE QLOCAL(ORANGE.LOCAL.QUEUE) REPLACE +  
DESCR(' ') +  
PUT(ENABLED) +  
DEFPRTY(0) +  
DEFPSIST(NO) +  
GET(ENABLED) +  
MAXDEPTH(5000) +  
MAXMSGL(1024) +  
DEFSOPT(SHARED) +  
NOHARDENBO +  
USAGE(NORMAL) +  
NOTRIGGER;  
.   
.
```

Figure 32. Extrait du fichier de commandes MQSC, myprog.in

Pour la portabilité entre les environnements IBM MQ, limitez la longueur de ligne dans les fichiers de commandes MQSC à 72 caractères. Le signe plus indique que la commande se poursuit sur la ligne suivante.

Les attributs d'objet spécifiés dans MQSC sont affichés dans cette section en majuscules (par exemple, RQMNAME), bien qu'ils ne soient pas sensibles à la casse.

Remarque :

1. Le format d'un fichier MQSC ne dépend pas de son emplacement dans le système de fichiers.
2. Les noms d'attribut MQSC sont limités à huit caractères.
3. Les commandes MQSC sont disponibles sur d'autres plateformes, notamment z/OS.

Pour une description de chaque commande MQSC et de sa syntaxe, voir [«Commandes de script \(MQSC\)»](#), à la page 11.

Administration à l'aide de commandes PCF sous IBM i

L'objectif des commandes PCF (Programmable Command Format) IBM MQ est de permettre la programmation des tâches d'administration dans un programme d'administration. Vous pouvez ainsi créer des files d'attente et des définitions de processus, et modifier des gestionnaires de files d'attente à partir d'un programme.

Les commandes PCF couvrent la même gamme de fonctions que les commandes MQSC. Cependant, contrairement aux commandes MQSC, les commandes PCF et leurs réponses ne sont pas dans un format texte que vous pouvez lire.

Vous pouvez écrire un programme pour émettre des commandes PCF vers n'importe quel gestionnaire de files d'attente du réseau à partir d'un noeud unique. De cette manière, vous pouvez à la fois centraliser et automatiser les tâches d'administration.

Chaque commande PCF est une structure de données imbriquée dans la partie données d'application d'un message IBM MQ . Chaque commande est envoyée au gestionnaire de files d'attente cible à l'aide de la fonction MQI MQPUT de la même manière que tout autre message. Le serveur de commandes du gestionnaire de files d'attente qui reçoit le message l'interprète comme un message de commande et exécute la commande. Pour obtenir les réponses, l'application émet un appel MQGET et les données de réponse sont renvoyées dans une autre structure de données. L'application peut ensuite traiter la réponse et agir en conséquence.

En bref, voici quelques-uns des éléments que le programmeur d'application doit spécifier pour créer un message de commande PCF:

Descripteur de message

Il s'agit d'un descripteur de message IBM MQ standard, dans lequel:

- Le type de message (*MsgType*) est MQMT_REQUEST.
- Le format de message (*Format*) est MQFMT_ADMIN.

Données d'application

Contient le message PCF incluant l'en-tête PCF, dans lequel:

- Le type de message PCF (*Type*) indique MQCFT_COMMAND.
- L'identificateur de la commande indique la commande, par exemple *Change Queue* (MQCMD_CHANGE_Q).

Les fichiers PCF d'arrêt programme sont des commandes PCF qui contiennent des commandes MQSC dans le texte du message. Vous pouvez utiliser des fichiers PCF pour envoyer des commandes à un gestionnaire de files d'attente éloignées. Pour plus d'informations, voir [«Utilisation de MQAI pour simplifier l'utilisation des fichiers PCF»](#), à la page 35.

Pour obtenir une description complète des structures de données PCF et savoir comment les implémenter, voir [Structures des commandes et des réponses](#).

Utilisation du IBM MQ Explorer avec IBM MQ for IBM i

Utilisez ces informations pour administrer IBM MQ for IBM i à l'aide de la IBM MQ Explorer.

IBM MQ for Windows (plateformex86) et IBM MQ for Linux (plateformesx86 et x86-64) fournissent une interface d'administration appelée IBM MQ Explorer pour effectuer des tâches d'administration au lieu d'utiliser des commandes CL, de contrôle ou MQSC.

Le IBM MQ Explorer vous permet d'effectuer l'administration locale ou à distance de votre réseau à partir d'un ordinateur exécutant Windows (plateformex86) ou Linux (plateformesx86 et x86-64), en pointant le IBM MQ Explorer sur les gestionnaires de files d'attente et les clusters qui vous intéressent.

Grâce à IBM MQ Explorer, vous pouvez :

- Démarrez et arrêtez un gestionnaire de files d'attente (sur votre machine locale uniquement).
- Définissez, affichez et modifiez les définitions des objets IBM MQ tels que les files d'attente, les rubriques et les canaux.

- Parcourez les messages d'une file d'attente.
- Démarrer et arrêter un canal.
- Afficher les informations de statut d'un canal.
- Afficher les gestionnaires de files d'attente dans un cluster.
- Vérifiez les applications, les utilisateurs ou les canaux pour lesquels une file d'attente particulière est ouverte.
- Créez un cluster de gestionnaires de files d'attente à l'aide de l'assistant **Créer un cluster** .
- Ajoutez un gestionnaire de files d'attente à un cluster à l'aide de l'assistant **Ajout d'un gestionnaire de files d'attente à un cluster** .
- Gérez l'objet d'informations d'authentification, utilisé avec la sécurité du canal TLS (Transport Layer Security).

En utilisant les conseils en ligne, vous pouvez:

- Définissez et contrôlez diverses ressources, notamment les gestionnaires de files d'attente, les files d'attente, les canaux, les définitions de processus, les canaux de connexion client, les programmes d'écoute, les rubriques, les services, les listes de noms et les clusters.
- Démarrer ou arrêter un gestionnaire de files d'attente et ses processus associés.
- Affichez les gestionnaires de files d'attente et les objets associés sur votre poste de travail ou à partir d'autres postes de travail.
- Vérifiez le statut des gestionnaires de files d'attente, des clusters et des canaux.

Vérifiez que vous avez satisfait les exigences suivantes avant de tenter d'utiliser IBM MQ Explorer pour gérer IBM MQ sur une machine serveur. Vérifiez que:

1. Un serveur de commandes est en cours d'exécution pour tous les gestionnaires de files d'attente administrés, démarrés sur le serveur par la commande CL **STRMQMCSVR**.
2. Un programme d'écoute TCP/IP approprié existe pour chaque gestionnaire de files d'attente éloignées. Il s'agit du programme d'écoute IBM MQ démarré par la commande **STRMQMLSR** .
3. Le canal de connexion serveur, appelé SYSTEM . ADMIN . SVRCONN, existe sur chaque gestionnaire de files d'attente éloignées. Vous devez créer ce canal vous-même. Il est obligatoire pour chaque gestionnaire de files d'attente éloignées administré. Sans elle, l'administration à distance n'est pas possible.
4. Vérifiez que la file d'attente SYSTEM . MQEXPLORER . REPLY . MODEL existe.

Gestion du serveur de commandes pour l'administration à distance sous IBM i

Utilisez ces informations pour en savoir plus sur l'administration à distance du serveur de commandes IBM MQ for IBM i .

Chaque gestionnaire de files d'attente peut être associé à un serveur de commandes. Un serveur de commandes traite toutes les commandes entrantes provenant des gestionnaires de files d'attente éloignées ou les commandes PCF provenant des applications. Il présente les commandes au gestionnaire de files d'attente pour le traitement et renvoie un code achèvement ou un message d'opérateur en fonction de l'origine de la commande.

Un serveur de commandes est obligatoire pour toute administration impliquant des fichiers PCF, MQAI, ainsi que pour l'administration à distance.

Remarque : Pour l'administration à distance, vous devez vous assurer que le gestionnaire de files d'attente cible est en cours d'exécution. Sinon, les messages contenant des commandes ne peuvent pas quitter le gestionnaire de files d'attente à partir duquel elles sont émises. A la place, ces messages sont mis en file d'attente dans la file d'attente de transmission locale qui sert le gestionnaire de files d'attente éloignées. Dans la mesure du possible, évitez cette situation.

Il existe des commandes de contrôle distinctes pour le démarrage et l'arrêt du serveur de commandes. Vous pouvez effectuer les opérations décrites dans les sections suivantes à l'aide de l'explorateur IBM MQ.

Démarrage et arrêt du serveur de commandes

Pour démarrer le serveur de commandes, utilisez la commande CL suivante:

```
STRMQMCSVR MQMNAME('saturn.queue.manager')
```

où `saturn.queue.manager` est le gestionnaire de files d'attente pour lequel le serveur de commandes est démarré.

Pour arrêter le serveur de commandes, utilisez l'une des commandes CL suivantes:

1.

```
ENDMQMCSVR MQMNAME('saturn.queue.manager') OPTION(*CNTRLD)
```

pour effectuer un arrêt contrôlé, où `saturn.queue.manager` correspond au gestionnaire de files d'attente pour lequel le serveur de commandes est arrêté. Il s'agit de l'option par défaut, qui signifie que `OPTION(*CNTRLD)` peut être omis.

2.

```
ENDMQMCSVR MQMNAME('saturn.queue.manager') OPTION(*IMMED)
```

pour effectuer un arrêt immédiat, où `saturn.queue.manager` correspond au gestionnaire de files d'attente pour lequel le serveur de commandes est arrêté.

Affichage de l'état du serveur de commandes

Pour l'administration à distance, vérifiez que le serveur de commandes sur le gestionnaire de files d'attente cible est en cours d'exécution. S'il n'est pas en cours d'exécution, les commandes distantes ne peuvent pas être traitées. Tous les messages contenant des commandes sont mis en file d'attente dans la file d'attente de commandes du gestionnaire de files d'attente cible `SYSTEM.ADMIN.COMMAND.QUEUE`.

Pour afficher l'état du serveur de commandes d'un gestionnaire de files d'attente, appelé ici `saturn.queue.manager`, la commande CL est la suivante:

```
DSPMQMCSVR MQMNAME('saturn.queue.manager')
```

Exécutez cette commande sur la machine cible. Si le serveur de commandes est en cours d'exécution, le panneau présenté dans la [Figure 33](#), à la page 275 apparaît:

```
Display MQM Command Server (DSPMQMCSVR)

Queue manager name . . . . . > saturn.queue.manager
MQM Command Server Status. . . . > RUNNING

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

Figure 33. Afficher le panneau MQM Command Server

IBM i Gestion des travaux pour IBM i

Ces informations décrivent la manière dont IBM MQ traite les demandes de travail et détaillent les options disponibles pour hiérarchiser et contrôler les travaux associés à IBM MQ.

Avertissement

Ne modifiez pas les objets de gestion des travaux IBM MQ à moins que vous ne compreniez parfaitement les concepts de la gestion des travaux IBM i et IBM MQ .

Des informations supplémentaires sur les sous-systèmes et les descriptions de travail sont disponibles sous [Gestion des travaux](#) dans la documentation du produit IBM i . Prêtez une attention particulière aux sections [Démarrage des travaux](#) et [Travaux par lots](#).

IBM MQ for IBM i intègre l'environnement IBM i UNIX et les unités d'exécution IBM i . N'apportez **pas** de modifications aux objets du Integrated File System (IFS).

Lors des opérations normales, un gestionnaire de files d'attente IBM MQ démarre un certain nombre de travaux par lots pour effectuer différentes tâches. Par défaut, ces travaux par lots s'exécutent dans le sous-système QMQM créé lors de l'installation de IBM MQ .

La gestion des travaux fait référence au processus de personnalisation des tâches IBM MQ pour obtenir les performances optimales de votre système ou pour simplifier l'administration.

Par exemple, vous pouvez effectuer les tâches suivantes :

- Modifiez la priorité d'exécution des travaux pour rendre un gestionnaire de files d'attente plus réactif qu'un autre.
- Redirige la sortie d'un certain nombre de travaux vers une file d'attente en sortie spécifique.
- Faites en sorte que tous les travaux d'un certain type soient exécutés dans un sous-système spécifique.
- Isolez les erreurs dans un sous-système.

La gestion des travaux s'effectue en créant ou en modifiant les descriptions de travail associées aux travaux IBM MQ . Vous pouvez configurer la gestion des travaux pour:

- Une installation IBM MQ complète.
- Gestionnaires de files d'attente individuels.
- Travaux individuels pour des gestionnaires de files d'attente individuels.

IBM i Tâches IBM MQ pour IBM i

Il s'agit d'un tableau des travaux IBM MQ for IBM i et d'une brève description de chacun d'eux.

Lorsqu'un gestionnaire de files d'attente est en cours d'exécution, vous voyez certains ou tous les travaux par lots suivants s'exécutant sous le profil utilisateur QMQM dans le sous-système IBM MQ . Les travaux sont décrits brièvement dans [Tableau 19](#), à la page 276.

Vous pouvez afficher tous les travaux connectés à un gestionnaire de files d'attente à l'aide de l'option 22 du panneau **Work with Queue Manager** (WRKMQM). Vous pouvez afficher les programmes d'écoute à l'aide de la commande WRKMQLSR.

| Nom du travail | Function |
|----------------|--|
| AMQALMPX | Processeur de point de contrôle qui prend régulièrement des points de contrôle de journal. |
| AMQZMUCO | Gestionnaire d'utilitaires. Ce travail exécute des utilitaires de gestionnaire de files d'attente critiques, par exemple le gestionnaire de chaînes de journaux. |

Tableau 19. Tâches IBM MQ . (suite)

| Nom du travail | Function |
|----------------|---|
| AMQZXMA0 | Contrôleur d'exécution correspondant au premier travail démarré par le gestionnaire de files d'attente. Il traite les demandes MQCONN et démarre les processus d'agent pour traiter les appels API IBM MQ . |
| AMQZFUMA | Gestionnaire des droits d'accès aux objets (OAM). |
| AMQZLAA0 | Les agents de gestionnaire de files d'attente qui effectuent la plupart du travail pour les applications qui se connectent au gestionnaire de files d'attente à l'aide de MQCNO_STANDARD_BINDING. |
| AMQZLSA0 | Agent de gestionnaire de files d'attente. |
| AMQZMUFO | Gestionnaire d'utilitaire |
| AMQZMGRO | Contrôleur de processus. Ce travail est utilisé pour démarrer et gérer les programmes d'écoute et les services. |
| AMQZMUR0 | Gestionnaire d'utilitaires. Ce travail exécute des utilitaires de gestionnaire de files d'attente critiques, par exemple le gestionnaire de chaînes de journaux. |
| AMQFQPUB | Démon de publication / abonnement en file d'attente. |
| AMQFCXBA | Travail d'agent de courtier. |
| RUNMQBRK | Travail de contrôle du courtier. |
| AMQRMPPA | Travail de regroupement de processus de canal. |
| amqcrsta | Canal répondeur appelé par TCP/IP. |
| AMQCRS6B | Canal récepteur LU62 et connexion client (voir remarque). |
| AMQRRMFA | Gestionnaire de référentiels pour les clusters. |
| AMQCLMAA | Programme d'écoute TCP/IP sans unités d'exécution. |
| AMQPCSEA | Processeur de commandes PCF qui gère les demandes PCF et d'administration à distance. |
| runmqtrm | Moniteur de déclenchement. |
| runmqdlq | Gestionnaire de file d'attente de rebut. |
| runmqchi | Initiateur de canal. |
| runmqchl | Travail de canal émetteur démarré pour chaque canal émetteur. |
| runmqlsr | Programme d'écoute TCP/IP à unités d'exécution. |
| AMQRCMLA | Canal MQSC et processeur de commandes PCF. |

Remarque : Le travail récepteur LU62 s'exécute dans le sous-système de communications et extrait ses propriétés d'exécution des entrées de routage et de communication utilisées pour démarrer le travail. Pour plus d'informations, voir [Initiated end \(Receiver\)](#) .

Objets de gestion des travaux sous IBM i

Lorsque IBM MQ est installé, divers objets sont fournis dans la bibliothèque QMQM pour faciliter la gestion des travaux. Ces objets sont ceux dont les travaux IBM MQ ont besoin pour s'exécuter dans leur propre sous-système.

Des exemples de descriptions de travail sont fournis pour deux des travaux par lots IBM MQ . Si aucune description de travail spécifique n'est fournie pour un travail IBM MQ , elle s'exécute avec la description de travail par défaut QMQMJOB.

Les objets de gestion des travaux qui sont fournis lorsque vous installez IBM MQ sont répertoriés dans [Tableau 20](#), à la page 278 et les objets créés pour un gestionnaire de files d'attente sont répertoriés dans [Tableau 21](#), à la page 278

Remarque : Les objets de gestion des travaux se trouvent dans la bibliothèque QMQM et les objets de gestionnaire de files d'attente se trouvent dans la bibliothèque du gestionnaire de files d'attente.

| Nom | Tapez | Description |
|-----------|-------|---|
| AMQALMPX | *JOB | Description de travail utilisée par le processus de point de contrôle |
| AMQZLAA0 | *JOB | Description de travail utilisée par les processus de l'agent IBM MQ |
| AMQZLSA0 | *JOB | Agent de gestionnaire de files d'attente de liaisons isolées |
| AMQZXMA0 | *JOB | Description de travail utilisée par les contrôleurs d'exécution IBM MQ |
| QMQM | *SBS | Sous-système dans lequel tous les travaux IBM MQ s'exécutent |
| QMQM | *JOB | File d'attente de travaux connectée au sous-système fourni |
| QMQMJOB | *JOB | Description de travail IBM MQ par défaut, utilisée s'il n'existe pas de description de travail spécifique pour un travail |
| QMQMSG | *MSG | File d'attente de messages par défaut pour les travaux IBM MQ . |
| QMQMRUN20 | *CLS | Description de classe pour les travaux IBM MQ à priorité élevée |
| QMQMRUN35 | *CLS | Description de classe pour les travaux IBM MQ de priorité moyenne |
| QMQMRUN50 | *CLS | Description de classe pour les travaux IBM MQ de priorité basse |

| Nom | Tapez | Description |
|------------|-------|---|
| AMQA000000 | *JRN | Récepteur de journal local |
| AMQAJRN | *JRN | Journal local |
| AMQJRNINF | *USR | Espace utilisateur mis à jour avec les derniers récepteurs de journal requis pour le démarrage et la reprise sur support d'un gestionnaire de files d'attente. Cet espace utilisateur peut être interrogé par une application pour déterminer quels récepteurs de journal doivent être archivés et lesquels peuvent être supprimés en toute sécurité. |
| AMQAJRNMSG | *MSG | File d'attente de messages du journal local |
| AMQCRC6B | *PGM | Programme de démarrage de la connexion LU6.2 |

Tableau 21. Objets de gestion des travaux créés pour un gestionnaire de files d'attente (suite)

| Nom | Tapez | Description |
|----------|-------|---|
| AMQRFOLD | *FILE | Fichier de définition de canal de gestionnaire de files d'attente migré |
| QMQMMSG | *MSGQ | File de messages du gestionnaire de files d'attente |

IBM i Comment IBM MQ utilise les objets de gestion des travaux dans IBM i

Ces informations décrivent la manière dont IBM MQ utilise les objets de gestion des travaux et fournissent des exemples de configuration.



Avvertissement : Ne modifiez pas les paramètres des postes de file d'attente de travaux dans le sous-système QMQM pour limiter le nombre de travaux autorisés dans le sous-système par priorité. Si vous tentez d'effectuer cette opération, vous pouvez arrêter l'exécution des travaux IBM MQ essentiels une fois qu'ils ont été soumis et provoquer l'échec du démarrage du gestionnaire de files d'attente.

Pour comprendre comment configurer la gestion des travaux, vous devez d'abord comprendre comment IBM MQ utilise les descriptions de travail.

La description de travail utilisée pour démarrer le travail contrôle de nombreux attributs du travail. Exemple :

- File d'attente de travaux dans laquelle le travail est mis en file d'attente et sur quel sous-système le travail est exécuté.
- Données de routage utilisées pour démarrer le travail et la classe utilisés par le travail pour ses paramètres d'exécution.
- File d'attente en sortie utilisée par le travail pour les fichiers d'impression.

Le processus de démarrage d'un travail IBM MQ peut être considéré en trois étapes:

1. IBM MQ sélectionne une description de travail.

IBM MQ utilise la technique suivante pour déterminer la description de travail à utiliser pour un travail par lots:

- Recherchez dans la bibliothèque du gestionnaire de files d'attente une description de travail portant le même nom que le travail. Voir [Présentation des noms de bibliothèque de gestionnaire de files d'attente IBM MQ for IBM i](#) pour plus de détails sur la bibliothèque de gestionnaire de files d'attente.
- Recherchez la description de travail par défaut QMQMJOB dans la bibliothèque du gestionnaire de files d'attente.
- Recherchez dans la bibliothèque QMQM une description de travail portant le même nom que le travail.
- Utilisez la description de travail par défaut, QMQMJOB, dans la bibliothèque QMQM.

2. Le travail est soumis à la file d'attente de travaux.

Les descriptions de travail fournies avec IBM MQ ont été configurées, par défaut, pour placer des travaux dans la file d'attente de travaux QMQM dans la bibliothèque QMQM. La file d'attente de travaux QMQM étant connectée au sous-système QMQM fourni, les travaux démarrent par défaut dans le sous-système QMQM.

3. Le travail entre dans le sous-système et passe par les phases d'activation.

Lorsque le travail entre dans le sous-système, les données de routage indiquées dans la description de travail sont utilisées pour rechercher des postes de routage pour le travail.

Les données de routage doivent correspondre à l'une des entrées de routage définies dans le sous-système QMQM, ce qui définit les classes fournies (QMQRUN20, QMQRUN35 ou QMQRUN50) qui sont utilisées par le travail.

Remarque : Si les travaux IBM MQ ne semblent pas démarrer, assurez-vous que le sous-système est en cours d'exécution et que la file d'attente de travaux n'est pas suspendue,

Si vous avez modifié les objets de gestion des travaux IBM MQ, assurez-vous que tout est correctement associé. Par exemple, si vous spécifiez une file d'attente de travaux autre que QMQM/QMQM dans la description de travail, assurez-vous qu'une opération ADDJOBQE est effectuée pour le sous-système, c'est-à-dire QMQM.

Vous pouvez créer une description de travail pour chaque travail documenté dans [Tableau 19](#), à la page 276 en utilisant la feuille de travail suivante comme exemple:

```
What is the queue manager library name? _____
Does job description AMQZXMA0 exist in the queue manager library? Yes No
Does job description QMQMJOB0 exist in the queue manager library? Yes No
Does job description AMQZXMA0 exist in the QMQM library? Yes No
Does job description QMQMJOB0 exist in the QMQM library? Yes No
```

Si vous répondez Non à toutes ces questions, créez une description de travail globale QMQMJOB0 dans la bibliothèque QMQM.

File d'attente de messages IBM MQ

Une file d'attente de messages IBM MQ, QMQMMSG, est créée dans chaque bibliothèque de gestionnaire de files d'attente. Les messages du système d'exploitation sont envoyés à cette file d'attente lorsque les travaux du gestionnaire de files d'attente s'arrêtent et IBM MQ envoie des messages à la file d'attente. Par exemple, pour signaler les récepteurs de journal nécessaires au démarrage. Conservez le nombre de messages dans cette file d'attente de messages à une taille gérable afin de faciliter la surveillance.

IBM i

Exemples de système par défaut pour IBM i

Ces exemples montrent comment fonctionne une installation IBM MQ non modifiée lorsque certains des travaux standard sont soumis au démarrage du gestionnaire de files d'attente.

Tout d'abord, le travail du contrôleur d'exécution AMQZXMA0 démarre.

1. Exécutez la commande **STRMQM** pour le gestionnaire de files d'attente TESTQM.
2. IBM MQ recherche dans la bibliothèque du gestionnaire de files d'attente QMTESTQM, d'abord la description de travail AMQZXMA0, puis la description de travail QMQMJOB0.

Comme aucune de ces descriptions de travail n'existe, IBM MQ recherche la description de travail AMQZXMA0 dans la bibliothèque de produits QMQM. Cette description de travail existe, elle est donc utilisée pour soumettre le travail.

3. La description de travail utilise la file d'attente de travaux par défaut IBM MQ, de sorte que le travail est soumis à la file d'attente de travaux QMQM/QMQM.
4. Les données de routage de la description de travail AMQZXMA0 étant QMQRUN20, le système recherche les entrées de routage de sous-système correspondant à ces données.

Par défaut, le poste de routage avec le numéro de séquence 9900 possède des données de comparaison qui correspondent à QMQRUN20, de sorte que le travail est démarré avec la classe définie sur ce poste de routage, également appelé QMQRUN20.

5. La priorité d'exécution de la classe QMQM/QMQRUN20 étant définie sur 20, le travail AMQZXMA0 s'exécute dans le sous-système QMQM avec la même priorité que la plupart des travaux interactifs sur le système.

Ensuite, le travail du processus de point de contrôle AMQALMPX démarre.

1. IBM MQ recherche dans la bibliothèque du gestionnaire de files d'attente QMTESTQM, d'abord la description de travail AMQALPMX, puis la description de travail QMQMJOB0.

Aucune de ces descriptions de travail n'existe. Par conséquent, IBM MQ recherche les descriptions de travail AMQALMPX et QMQMJOBDD dans la bibliothèque de produit QMQM.

La description de travail AMQALMPX n'existe pas, contrairement à QMQMJOBDD. QMQMJOBDD est donc utilisé pour soumettre le travail.

Remarque : La description de travail QMQMJOBDD est toujours utilisée pour les travaux IBM MQ qui n'ont pas leur propre description de travail.

2. La description de travail utilise la file d'attente de travaux par défaut IBM MQ , de sorte que le travail est soumis à la file d'attente de travaux QMQM/QMQM.
3. Les données de routage de la description de travail QMQMJOBDD étant QMQMRUN35, le système recherche les entrées de routage de sous-système correspondant à ces données.

Par défaut, l'entrée de routage avec le numéro de séquence 9910 possède des données de comparaison qui correspondent à QMQMRUN35, de sorte que le travail est démarré avec la classe définie sur cette entrée de routage, également appelée QMQMRUN35.

4. La priorité d'exécution de la classe QMQM/QMQMRUN35 étant définie sur 35, le travail AMQALMPX s'exécute dans le sous-système QMQM avec une priorité inférieure à celle de la plupart des travaux interactifs du système, mais avec une priorité supérieure à celle de la plupart des travaux par lots.

IBM i

Configuration d'exemples de gestion des travaux sous IBM i

Utilisez ces informations pour savoir comment modifier et créer des descriptions de travail IBM MQ afin de modifier les attributs d'exécution des travaux IBM MQ .

La clé de la flexibilité de la gestion des travaux IBM MQ réside dans la manière à deux niveaux dont IBM MQ recherche les descriptions de travail:

- Si vous créez ou modifiez des descriptions de travail dans une bibliothèque de gestionnaire de files d'attente, ces modifications remplacent les descriptions de travail globales dans QMQM, mais les modifications sont locales et affectent uniquement ce gestionnaire de files d'attente particulier.
 - Si vous créez ou modifiez des descriptions de travail globales dans la bibliothèque QMQM, ces descriptions de travail affectent tous les gestionnaires de files d'attente du système, sauf si elles sont remplacées localement pour des gestionnaires de files d'attente individuels.
1. L'exemple suivant augmente la priorité des travaux de contrôle de canal pour un gestionnaire de files d'attente individuel.

Pour que les travaux du gestionnaire de référentiels et de l'initiateur de canal, AMQRRMFA et RUNMQCHI, soient exécutés aussi rapidement que possible pour le gestionnaire de files d'attente TESTQM, procédez comme suit:

- a. Créez des doublons locaux de la description de travail QMQM/QMQMJOBDD avec les noms des processus IBM MQ que vous souhaitez contrôler dans la bibliothèque du gestionnaire de files d'attente. Exemple :

```
CRTDUPOBJ OBJ(QMQMJOBDD) FROMLIB(QMQM) OBJTYPE(*JOBDD) TOLIB(QMTESTQM)
NEWOBJ (RUNMQCHI)
CRTDUPOBJ OBJ(QMQMJOBDD) FROMLIB(QMQM) OBJTYPE(*JOBDD) TOLIB(QMTESTQM)
NEWOBJ (AMQRRMFA)
```

- b. Modifiez le paramètre de données de routage dans la description de travail pour vous assurer que les travaux utilisent la classe QMQMRUN20 .

```
CHGJOBDD JOBDD(QMTESTQM/RUNMQCHI) RTGDTA('QMQMRUN20')
CHGJOBDD JOBDD(QMTESTQM/AMQRRMFA) RTGDTA('QMQMRUN20')
```

Les travaux AMQRRMFA et RUNMQCHI pour le gestionnaire de files d'attente TESTQM sont maintenant:

- Utiliser les nouvelles descriptions de travail locales dans la bibliothèque du gestionnaire de files d'attente
 - S'exécute avec la priorité 20, car la classe QMQMRUN20 est utilisée lorsque les travaux entrent dans le sous-système.
2. L'exemple suivant définit une nouvelle classe de priorité d'exécution pour le sous-système QMQM.

- a. Créez une classe en double dans la bibliothèque QMQM pour permettre à d'autres gestionnaires de files d'attente d'accéder à la classe, en exécutant la commande suivante:

```
CRTDUPOBJ OBJ(QMQMRUN20) FROMLIB(QMQM) OBJTYPE(*CLS) TOLIB(QMQM)
NEWOBJ(QMQMRUN10)
```

- b. Modifiez la classe pour qu'elle ait la nouvelle priorité d'exécution en exécutant la commande suivante:

```
CHGCLS CLS(QMQM/QMQMRUN10) RUNPTY(10)
```

- c. Ajoutez la nouvelle définition de classe au sous-système en exécutant la commande suivante:

```
ADDRTGE SBS(QMQM/QMQM) SEQNBR(8999) CMPVAL('QMQMRUN10') PGM(QSYS/QCMD)
CLS(QMQM/QMQMRUN10)
```

Remarque : Vous pouvez indiquer n'importe quelle valeur numérique pour le numéro de séquence de routage, mais les valeurs doivent être séquentielles. Ce numéro de séquence indique au sous-système l'ordre dans lequel les postes de routage doivent être recherchés pour une correspondance de données de routage.

- d. Modifiez la description de travail locale ou globale pour utiliser la nouvelle classe de priorité en exécutant la commande suivante:

```
CHGJOB JOB(QMQMlibname/QMQMJOB) RTGDTA('QMQMRUN10')
```

Tous les travaux de gestionnaire de files d'attente associés à QMlibraryname utilisent désormais une priorité d'exécution de 10.

3. L'exemple suivant exécute un gestionnaire de files d'attente dans son propre sous-système

Pour que tous les travaux du gestionnaire de files d'attente TESTQM soient exécutés dans le sous-système QBATCH, procédez comme suit:

- a. Créez un doublon local de la description de travail QMQM/QMQMJOB dans la bibliothèque du gestionnaire de files d'attente à l'aide de la commande

```
CRTDUPOBJ OBJ(QMQMJOB) FROMLIB(QMQM) OBJTYPE(*JOB) TOLIB(QMTESTQM)
```

- b. Modifiez le paramètre de file d'attente de travaux dans la description de travail pour vous assurer que les travaux utilisent la file d'attente de travaux QBATCH.

```
CHGJOB JOB(QMTESTQM/QMQMJOB) JOBQ(*LIBL/QBATCH)
```

Remarque : La file d'attente de travaux est associée à la description de sous-système. Si vous constatez que les travaux restent dans la file d'attente de travaux, vérifiez que la définition de la file d'attente de travaux est définie sur le SBS. Utilisez la commande DSPSBS pour le sous-système et choisissez l'option 6, Postes de file d'attente de travaux.

Tous les travaux du gestionnaire de files d'attente TESTQM sont maintenant:

- Utiliser la nouvelle description de travail par défaut locale dans la bibliothèque du gestionnaire de files d'attente
- Sont soumis à la file d'attente de travaux QBATCH.

Pour vous assurer que les travaux sont routés et hiérarchisés correctement:

- Créez des entrées de routage pour les travaux IBM MQ dans le sous-système QBATCH ou
- Utilisez une entrée de routage fourre-tout qui appelle QCMD, quelles que soient les données de routage utilisées.

Cette option ne fonctionne que si l'option Nombre maximal de travaux actifs pour la file d'attente de travaux QBATCH est définie sur *NOMAX. La valeur par défaut du système est 1.

4. L'exemple suivant crée un autre sous-système IBM MQ

- a. Créez un sous-système en double dans la bibliothèque QMQM en exécutant la commande suivante:

```
CRTDUPOBJ OBJ(QMQM) FROMLIB(QMQM) OBJTYPE(*SBSD) TOLIB(QMQM) NEWOBJ(QMQM2)
```

- b. Supprimez la file d'attente de travaux QMQM en exécutant la commande suivante:

```
RMVJOBQE SBSB(QMQM/QMQM2) JOBQ(QMQM/QMQM)
```

- c. Créez une file d'attente de travaux pour le sous-système en exécutant la commande suivante:

```
CRTJOBQ JOBQ(QMQM/QMQM2) TEXT('Job queue for IBM MQ Queue Manager')
```

- d. Ajoutez un poste file d'attente de travaux au sous-système en exécutant la commande suivante:

```
ADDJOBQE SBSB(QMQM/QMQM2) JOBQ(QMQM/QMQM2) MAXACT(*NOMAX)
```

- e. Créez un QMQMJOBQ en double dans la bibliothèque du gestionnaire de files d'attente en exécutant la commande suivante:

```
CRTDUPOBJ OBJ(QMQMJOBQ) FROMLIB(QMQM) OBJTYPE(*JOBQ) TOLIB(QMlibraryname)
```

- f. Modifiez la description de travail pour utiliser la nouvelle file d'attente de travaux à l'aide de la commande suivante:

```
CHGJOBQ JOBQ(QMlibraryname/QMQMJOBQ) JOBQ(QMQM/QMQM2)
```

- g. Démarrez le sous-système en exécutant la commande suivante:

```
STRSBS SBSB(QMQM/QMQM2)
```

Remarque :

- a. Vous pouvez spécifier le sous-système dans n'importe quelle bibliothèque. Si, pour une raison quelconque, le produit est réinstallé ou que la bibliothèque QMQM est remplacée, les modifications apportées sont supprimées.
 - b. Tous les travaux de gestionnaire de files d'attente associés à QMlibraryname sont désormais exécutés sous le sous-système QMQM2.
5. L'exemple suivant collecte toutes les sorties pour un type de travail.

Pour collecter tous les processus de point de contrôle, AMQALMPX, les journaux de travaux de plusieurs gestionnaires de files d'attente dans une même file d'attente de sortie, procédez comme suit:

- a. Créez une file d'attente en sortie, par exemple

```
CRTOUTQ OUTQ(MYLIB/CHCKPTLOGS)
```

- b. Créez un doublon global de la description de travail QMQM/QMQMJOB, à l'aide du nom du processus IBM MQ que vous souhaitez contrôler, par exemple

```
CRTDUPOBJ OBJ(QMQMJOB) FROMLIB(QMQM) OBJTYPE(*JOB) NEWOBJ(AMQALMPX)
```

- c. Modifiez le paramètre de file d'attente en sortie de la description de travail pour qu'il pointe vers votre nouvelle file d'attente en sortie et modifiez le niveau de consignation du travail de sorte que tous les messages soient consignés dans l'historique du travail.

```
CHGJOB JOB(QMQM/AMQALMPX) OUTQ(MYLIB/CHKPTLOGS) LOG(4 00 *SECLVL)
```

Tous les travaux IBM MQ AMQALMPX, pour tous les gestionnaires de files d'attente, utilisent la nouvelle description de travail globale AMQALMPX, à condition qu'il n'y ait pas de descriptions de travail de remplacement locales dans la bibliothèque du gestionnaire de files d'attente local.

Tous les fichiers spoule de l'historique de ces travaux sont maintenant écrits dans la file d'attente en sortie CHKPTLOGS de la bibliothèque MYLIB.

Remarque :

- a. L'exemple précédent ne fonctionne que si QPJOBLOG, ou tout fichier d'impression, a la valeur *JOB pour son paramètre de file d'attente en sortie. Dans l'exemple précédent, le fichier QSYS/QPDJOBLOG doit avoir la valeur *JOB pour OUTQ.
- b. Pour modifier un fichier d'impression système, utilisez la commande CHGPRTF. Exemple :

```
CHGPRTF PRTF(QJOBLOG) OUTQ(*JOB)
```

L'option *JOB indique que les descriptions de travail doivent être utilisées.

- c. Vous pouvez envoyer tous les fichiers spoule associés aux travaux IBM MQ vers une file d'attente en sortie particulière. Cependant, vérifiez que le fichier d'impression utilisé a la valeur appropriée pour le paramètre OUTQ.

IBM i

Disponibilité, sauvegarde, récupération et redémarrage sur IBM i

Utilisez ces informations pour comprendre comment IBM MQ for IBM i utilise le support de journalisation IBM i pour aider sa stratégie de sauvegarde et de restauration.

Avant de lire cette section, vous devez connaître les méthodes de sauvegarde et de reprise IBM i standard, ainsi que l'utilisation des journaux et des récepteurs de journal associés sous IBM i. Pour plus d'informations sur ces rubriques, voir [Sauvegarde et reprise](#).

Pour comprendre la stratégie de sauvegarde et de reprise, vous devez d'abord comprendre comment IBM MQ for IBM i organise ses données dans le système de fichiers IBM i et le système de fichiers intégré (IFS).

IBM MQ for IBM i stocke ses données dans une bibliothèque individuelle pour chaque instance de gestionnaire de files d'attente et dans les fichiers STREAM du système de fichiers IFS.

Les bibliothèques spécifiques du gestionnaire de files d'attente contiennent les journaux, les récepteurs de journal et les objets requis pour contrôler la gestion des travaux du gestionnaire de files d'attente. Les répertoires et fichiers IFS contiennent les fichiers de configuration IBM MQ, les descriptions des objets IBM MQ et les données qu'ils contiennent.

Chaque modification apportée à ces objets, qui peut être récupérée en cas de défaillance du système, est enregistrée dans un journal *avant* qu'elle ne soit appliquée à l'objet approprié. Cela a pour effet que ces modifications peuvent être récupérées en réalisant les informations enregistrées dans le journal.

Vous pouvez configurer IBM MQ for IBM i pour utiliser plusieurs instances de gestionnaire de files d'attente sur différents serveurs afin d'augmenter la disponibilité du gestionnaire de files d'attente et d'accélérer la reprise en cas de défaillance d'un serveur ou d'un gestionnaire de files d'attente.

Utilisez ces informations pour comprendre comment IBM MQ for IBM i utilise les journaux dans son fonctionnement pour contrôler les mises à jour des objets locaux.

Chaque bibliothèque de gestionnaire de files d'attente contient un journal pour ce gestionnaire de files d'attente et le journal porte le nom QM *GRLIB*/AMQ *A* JRN, où QM *GRLIB* est le nom de la bibliothèque de gestionnaire de files d'attente et *A* est une lettre, A dans le cas d'un gestionnaire de files d'attente à instance unique, qui est unique pour l'instance de gestionnaire de files d'attente.

QM *GRLIB* prend le nom QM, suivi du nom du gestionnaire de files d'attente dans un format unique. Par exemple, un gestionnaire de files d'attente nommé TEST possède une bibliothèque de gestionnaire de files d'attente nommée QMTEST. La bibliothèque du gestionnaire de files d'attente peut être spécifiée lors de la création d'un gestionnaire de files d'attente à l'aide de la commande **CRTMQM**.

Les journaux sont associés à des récepteurs de journal qui contiennent les informations en cours de journalisation. Les récepteurs sont des objets auxquels les informations peuvent uniquement être ajoutées et qui finiront par se remplir.

Les récepteurs de journal utilisent un espace disque précieux avec des informations obsolètes. Toutefois, vous pouvez placer les informations dans la mémoire permanente pour réduire ce problème. Un récepteur de journal est attaché au journal à un moment donné. Si le récepteur de journal atteint sa taille de seuil prédéterminée, il est détaché et remplacé par un nouveau récepteur de journal. Vous pouvez spécifier le seuil des récepteurs de journal lorsque vous créez un gestionnaire de files d'attente à l'aide de **CRTMQM** et du paramètre **THRESHOLD**.

Les récepteurs de journal associés au journal IBM MQ for IBM i local existent dans chaque bibliothèque de gestionnaire de files d'attente et adoptent une convention de dénomination comme suit:

AMQ *Arnnnnn*

Où

a

est une lettre A-Z. Il s'agit de A pour les gestionnaires de files d'attente à instance unique. Il varie selon les instances d'un gestionnaire de files d'attente multi-instance.

nnnnn

est la valeur décimale 00000 to 99999 qui est incrémentée de 1 pour le journal suivant dans la séquence.

r

est la valeur décimale 0 to 9, qui est incrémentée de 1 chaque fois qu'un récepteur est restauré.

La séquence des journaux est basée sur la date. Toutefois, la désignation du journal suivant est basée sur les règles suivantes:

1. AMQA \pm nnnnn passe à AMQA \pm (nnnnn+1) et nnnnn effectue un retour à la ligne lorsqu'il atteint 99999. Par exemple, AMQA099999 va à AMQA000000 et AMQA999999 va à AMQA900000.
2. Si un journal portant un nom généré par la règle 1 existe déjà, le message CPI70E3 est envoyé à la file d'attente de messages QSYSOPR et la commutation automatique de récepteurs s'arrête.

Le récepteur actuellement attaché continue d'être utilisé jusqu'à ce que vous examiniez l'incident et que vous attaquiez manuellement un nouveau récepteur.

3. Si aucun nouveau nom n'est disponible dans la séquence (c'est-à-dire que tous les noms de journal possibles se trouvent sur le système), vous devez effectuer les deux opérations suivantes:
 - a. Supprimez les journaux dont vous n'avez plus besoin (voir [«Gestion des journaux sous IBM i»](#), à la page 290).
 - b. Enregistrez les modifications de journal dans le récepteur de journal le plus récent à l'aide de (**RCDMQMIMG**) puis répétez l'étape précédente. Cela permet de réutiliser les anciens noms de récepteurs de journal.

Le journal AMQAJRN utilise l'option MNGRCV (*SYSTEM) pour permettre au système d'exploitation de modifier automatiquement les récepteurs de journal lorsque le seuil est atteint. Pour plus d'informations sur la façon dont le système gère les récepteurs, voir *IBM i Backup and Recovery*.

La valeur de seuil par défaut du récepteur de journal est de 100 000 Ko. Vous pouvez définir une valeur supérieure lorsque vous créez le gestionnaire de files d'attente. La valeur initiale de l'attribut LogReceiverSize est écrite dans la section LogDefaults du fichier mqs.ini.

Lorsqu'un récepteur de journal dépasse le seuil indiqué, il est détaché et un nouveau récepteur de journal est créé, héritant des attributs du récepteur précédent. Les modifications apportées aux attributs LogReceiverSize ou LogASP après la création d'un gestionnaire de files d'attente sont ignorées lorsque le système associe automatiquement un nouveau récepteur de journal.

Pour plus d'informations sur la configuration du système, voir [Modification des informations de configuration sur IBM i](#).

Si vous devez modifier la taille des récepteurs de journal après la création du gestionnaire de files d'attente, créez un nouveau récepteur de journal et définissez son propriétaire sur QMQM à l'aide des commandes suivantes:

```
CRTJRNRCV JRNRCV(QM GRLIB/AMQ Arnnnnn) THRESHOLD(xxxxxx) +  
TEXT('MQM LOCAL JOURNAL RECEIVER')  
CHGOBJOWN OBJ(QM GRLIB/AMQ Arnnnnn) OBJTYPE(*JRNRCV) NEWOWN(QMQM)
```

Où

QMGLIB

Est le nom de la bibliothèque de votre gestionnaire de files d'attente

a

Est l'identificateur d'instance (généralement A).

rnrrrrr

Récepteur de journal suivant dans la séquence d'appellation décrite précédemment

xxxxxx

Est le nouveau seuil de récepteur (en ko)

Remarque : La taille maximale du récepteur est régie par le système d'exploitation. Pour vérifier cette valeur, consultez le mot clé THRESHOLD de la commande **CRTJRNRCV**.

A présent, attachez le nouveau récepteur au journal AMQAJRN à l'aide de la commande suivante:

```
CHGJRN JRN(QMGLIB/AMQ A JRN) JRNRCV(QMGLIB/AMQ Annnnnn)
```

Pour plus de détails sur la gestion de ces récepteurs de journal, voir [«Gestion des journaux sous IBM i»](#), à la page 290.

IBM i *Utilisation du journal du gestionnaire de files d'attente sous IBM i*

Utilisez ces informations pour comprendre comment IBM MQ for IBM i utilise les journaux dans son fonctionnement pour contrôler les mises à jour des objets locaux.

Les mises à jour persistantes des files d'attente de messages sont effectuées en deux étapes. Les enregistrements représentant la mise à jour sont d'abord écrits dans le journal, puis le fichier de file d'attente est mis à jour.

Les récepteurs de journal peuvent donc devenir plus à jour que les fichiers de file d'attente. Pour s'assurer que le processus de redémarrage commence à partir d'un point cohérent, IBM MQ utilise des points de contrôle.

Un point de contrôle est un point dans le temps où l'enregistrement décrit dans le journal est identique à l'enregistrement dans la file d'attente. Le point de contrôle lui-même est constitué de la série d'enregistrements de journal nécessaires pour redémarrer le gestionnaire de files d'attente. Par exemple, l'état de toutes les transactions (c'est-à-dire les unités de travail) actives au moment du point de contrôle.

Les points de contrôle sont générés automatiquement par IBM MQ. Ils sont utilisés lorsque le gestionnaire de files d'attente démarre et s'arrête, et après un certain nombre d'opérations consignées.

Vous pouvez forcer un gestionnaire de files d'attente à prendre un point de contrôle en émettant la commande RCDMQMIMG sur tous les objets d'un gestionnaire de files d'attente et en affichant les résultats, comme suit:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME) DSPJRNDTA(*YES)
```

Lorsque les files d'attente traitent d'autres messages, l'enregistrement de point de contrôle devient incohérent avec l'état en cours des files d'attente.

Lorsque IBM MQ est redémarré, il localise le dernier enregistrement de point de contrôle dans le journal. Ces informations sont contenues dans le fichier de point de contrôle qui est mis à jour à la fin de chaque point de contrôle. L'enregistrement de point de contrôle représente le point de cohérence le plus récent entre le journal et les données. Les données de ce point de contrôle sont utilisées pour régénérer les files d'attente telles qu'elles existaient au moment du point de contrôle. Lorsque les files d'attente sont recrées, le journal est alors lu pour ramener les files d'attente à l'état dans lequel elles se trouvaient avant l'échec du système ou la fermeture.

Pour comprendre comment IBM MQ utilise le journal, prenez en compte le cas d'une file d'attente locale appelée TESTQ dans le gestionnaire de files d'attente TEST. Il est représenté par le fichier IFS:

```
/QIBM/UserData/mqm/qmgrs/TEST/queues
```

Si un message spécifié est inséré dans cette file d'attente, puis extrait de la file d'attente, les actions qui ont lieu sont présentées dans la figure [Figure 34](#), à la page 287.

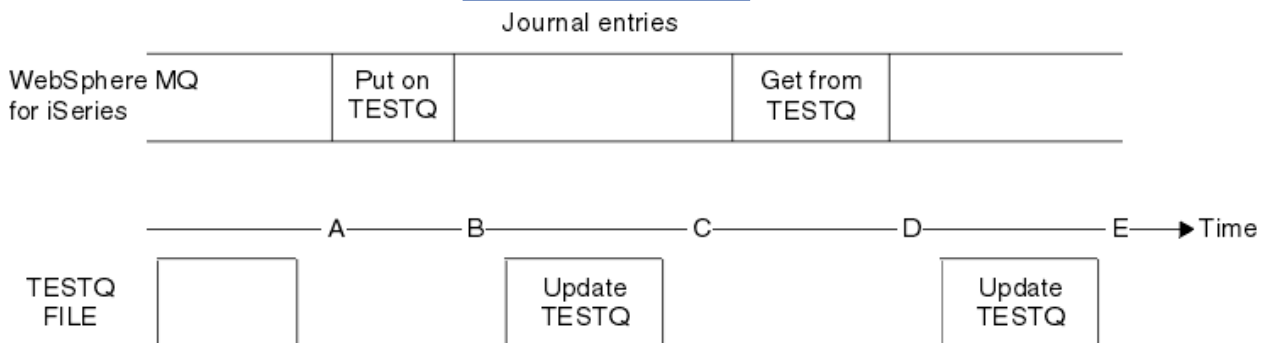


Figure 34. Séquence d'événements lors de la mise à jour des objets MQM

Les cinq points, de A à E, représentés dans le diagramme représentent des points dans le temps qui définissent les états suivants:

- A** La représentation du fichier IFS de la file d'attente est cohérente avec les informations contenues dans le journal.
- B** Un poste de journal est écrit dans le journal définissant une opération Put dans la file d'attente.
- C** La mise à jour appropriée est effectuée dans la file d'attente.
- D** Un poste de journal est écrit dans le journal définissant une opération Get à partir de la file d'attente.
- E** La mise à jour appropriée est effectuée dans la file d'attente.

La clé des fonctions de reprise de IBM MQ for IBM i est que l'utilisateur peut sauvegarder la représentation de fichier IFS de TESTQ comme à l'heure A, puis récupérer la représentation de fichier

IFS de TESTQ comme à l'heure E, en restaurant l'objet sauvegardé et en réalisant les entrées dans le journal à partir de l'heure A .

Cette stratégie est utilisée par IBM MQ for IBM i pour récupérer les messages persistants après une défaillance du système. IBM MQ se souvient d'une entrée particulière dans les récepteurs de journal et s'assure qu'au démarrage, il réexécute les entrées dans les journaux à partir de ce point. Cette entrée de démarrage est régulièrement recalculée afin que IBM MQ n'ait à effectuer que la relecture minimale nécessaire au prochain démarrage.

IBM MQ permet la récupération individuelle des objets. Toutes les informations persistantes relatives à un objet sont enregistrées dans les journaux IBM MQ for IBM i locaux. Tout objet IBM MQ endommagé ou endommagé peut être entièrement régénéré à partir des informations contenues dans le journal.

Pour plus d'informations sur la façon dont le système gère les récepteurs, voir [«Disponibilité, sauvegarde, récupération et redémarrage sur IBM i»](#), à la page 284.

IBM i *Images de support sous IBM i*

Sous IBM i, une image de support est une copie complète d'un objet IBM MQ qui est enregistré dans le journal. Certains objets endommagés ou endommagés peuvent être récupérés automatiquement à partir de leur image de support.

Un objet IBM MQ de longue durée peut représenter un grand nombre d'entrées de journal, remontant au point où il a été créé. Pour éviter cela, IBM MQ for IBM i a le concept d'image de support d'un objet.

Cette image de support est une copie complète de l'objet IBM MQ enregistré dans le journal. Si une image d'un objet est prise, l'objet peut être régénéré en réalisant les postes de journal à partir de cette image. Le poste du journal qui représente le point de relecture pour chaque objet IBM MQ est appelé poste de reprise sur incident lié au support. IBM MQ effectue le suivi des éléments suivants:

- Poste de reprise sur incident lié au support pour chaque objet gestionnaire de files d'attente.
- Entrée la plus ancienne de cet ensemble (voir le message d'erreur AMQ7462 dans [«Gestion des journaux sous IBM i»](#), à la page 290 pour plus de détails.

Les images de l'objet *CTLG et de l'objet *MQM sont prises régulièrement car ces objets sont essentiels au redémarrage du gestionnaire de files d'attente.

Des images d'autres objets sont prises lorsque cela vous convient. Par défaut, les images de tous les objets sont prises lorsqu'un gestionnaire de files d'attente est arrêté à l'aide de la commande **ENDMQM** avec le paramètre ENDCCTJOB (*YES). Cette opération peut prendre beaucoup de temps pour les gestionnaires de files d'attente de très grande taille. Si vous devez vous arrêter rapidement, indiquez le paramètre RCDMQMIMG (*NO) avec ENDCCTJOB (*YES). Dans ce cas, il est recommandé d'enregistrer une image de support complète dans les journaux après le redémarrage du gestionnaire de files d'attente, à l'aide de la commande suivante:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME)
```

IBM MQ enregistre automatiquement une image d'un objet, s'il trouve un point pratique où un objet peut être décrit de manière compacte par une petite entrée dans le journal. Toutefois, cela peut ne jamais se produire pour certains objets, par exemple, les files d'attente qui contiennent systématiquement un grand nombre de messages.

Plutôt que d'autoriser la date de l'entrée de reprise sur support la plus ancienne à se poursuivre pendant une période inutilement longue, utilisez la IBM MQ commande RCDMQMIMG, qui vous permet de prendre manuellement une image des objets sélectionnés.

Reprise à partir d'images de support

IBM MQ récupère automatiquement certains objets à partir de leur image de support s'il s'avère qu'ils sont endommagés ou endommagés. En particulier, cela s'applique aux objets spéciaux *MQM et *CTLG dans le cadre du démarrage normal du gestionnaire de files d'attente. Si une transaction de point de

synchronisation était incomplète au moment du dernier arrêt du gestionnaire de files d'attente, toute file d'attente affectée est également récupérée automatiquement afin de terminer l'opération de démarrage.

Vous devez récupérer d'autres objets manuellement à l'aide de la commande IBM MQ RCRMQM0BJ. Cette commande réexécute les entrées du journal pour recréer l'objet IBM MQ. Si un objet IBM MQ est endommagé, les seules actions valides sont de le supprimer ou de le recréer par cette méthode. Notez toutefois que les messages non persistants ne peuvent pas être récupérés de cette manière.

IBM i Points de contrôle sur IBM MQ for IBM i

Des points de contrôle sont pris à différents moments pour fournir un point de départ cohérent connu pour la reprise.

Le processus de point de contrôle AMQALMPX est chargé de prendre le point de contrôle aux points suivants:

- Démarrage du gestionnaire de files d'attente (STRMQM).
- Arrêt du gestionnaire de files d'attente (ENDMQM).
- Après un certain temps écoulé depuis le dernier point de contrôle (la période par défaut est de 30 minutes) et un nombre minimum d'enregistrements de journal ont été écrits depuis le point de contrôle précédent (la valeur par défaut est 100).
- Après l'écriture d'un certain nombre d'enregistrements de journal. La valeur par défaut est 10 000.
- Une fois que la taille de seuil du journal a été dépassée, un nouveau récepteur de journal a été créé automatiquement.
- Lorsqu'une image de support complète est prise avec:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(Q_MGR_NAME) DSPJRNDTA(*YES)
```

IBM i Sauvegardes des données IBM MQ for IBM i

Utilisez ces informations pour comprendre les deux types de sauvegarde IBM MQ pour chaque gestionnaire de files d'attente.

Pour chaque gestionnaire de files d'attente, il existe deux types de sauvegarde IBM MQ à prendre en compte:

- Sauvegarde des données et du journal.

Pour vous assurer que les deux ensembles de données sont cohérents, effectuez cette opération uniquement après avoir arrêté le gestionnaire de files d'attente.

- Sauvegarde du journal.

Vous pouvez effectuer cette opération lorsque le gestionnaire de files d'attente est actif.

Pour les deux méthodes, vous devez rechercher les noms du répertoire IFS du gestionnaire de files d'attente et de la bibliothèque du gestionnaire de files d'attente. Vous pouvez les trouver dans le fichier de configuration IBM MQ (mq5.ini). Pour plus d'informations, voir la [strophe QueueManager](#).

Utilisez les procédures suivantes pour effectuer les deux types de sauvegarde:

Sauvegarde des données et du journal d'un gestionnaire de files d'attente particulier

Remarque : N'utilisez pas de demande de sauvegarde en mise à jour lorsque le gestionnaire de files d'attente est en cours d'exécution. Une telle demande ne peut aboutir que si toutes les définitions de validation comportant des modifications en attente sont validées ou invalidées. Si cette commande est utilisée lorsque le gestionnaire de files d'attente est actif, les connexions de canal risquent de ne pas s'arrêter normalement. Utilisez toujours la procédure suivante.

1. Créez un récepteur de journal vide à l'aide de la commande suivante:

```
CHGJRN JRN(QMTEST/AMQAJRN) JRNRCV(*GEN)
```

- Utilisez la commande **RCDMQMIMG** pour enregistrer une image MQM pour tous les objets IBM MQ , puis forcez un point de contrôle à l'aide de la commande:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) MQMNAME(TEST)
```

- Arrêtez les canaux et vérifiez que le gestionnaire de files d'attente n'est pas en cours d'exécution. Si votre gestionnaire de files d'attente est en cours d'exécution, arrêtez-le à l'aide de la commande **ENDMQM** .
- Sauvegardez la bibliothèque du gestionnaire de files d'attente en exécutant la commande suivante:

```
SAVLIB LIB(QMTEST)
```

- Sauvegardez les répertoires IFS du gestionnaire de files d'attente en exécutant la commande suivante:

```
SAV DEV(...) OBJ((' /QIBM/UserData/mqm/qmgrs/test'))
```

Sauvegarde de journal d'un gestionnaire de files d'attente particulier

Etant donné que toutes les informations pertinentes sont conservées dans les journaux, tant que vous effectuez une sauvegarde intégrale à un moment donné, des sauvegardes partielles peuvent être effectuées en sauvegardant les récepteurs de journal. Elles enregistrent toutes les modifications apportées depuis l'heure de la sauvegarde intégrale et sont exécutées à l'aide des commandes suivantes:

- Créez un récepteur de journal vide à l'aide de la commande suivante:

```
CHGJRN JRN(QMTEST/AMQAJRN) JRNRCV(*GEN)
```

- Utilisez la commande **RCDMQMIMG** pour enregistrer une image MQM pour tous les objets IBM MQ , puis forcez un point de contrôle à l'aide de la commande:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) MQMNAME(TEST)
```

- Sauvegardez les récepteurs de journal à l'aide de la commande suivante:

```
SAVOBJ OBJ(AMQ*) LIB(QMTEST) OBJTYPE(*JRNRCV) .....
```

Une stratégie de sauvegarde simple consiste à effectuer une sauvegarde intégrale des bibliothèques IBM MQ toutes les semaines et à effectuer une sauvegarde quotidienne du journal. Cela dépend, bien sûr, de la manière dont vous avez configuré votre stratégie de sauvegarde pour votre entreprise.

Gestion des journaux sous IBM i

Dans le cadre de votre stratégie de sauvegarde, prenez soin de vos récepteurs de journal. Il est utile de supprimer les récepteurs de journal des bibliothèques IBM MQ pour diverses raisons:

- Pour libérer de l'espace ; s'applique à tous les récepteurs de journal
- Pour améliorer les performances lors du démarrage (STRMQM)
- Pour améliorer les performances de la création d'objets (RCRMQMOBJ)

Avant de supprimer un récepteur de journal, vous devez vous assurer que vous disposez d'une copie de sauvegarde et que vous n'en avez plus besoin.

Les récepteurs de journal peuvent être supprimés de la bibliothèque du gestionnaire de files d'attente *après* avoir été déconnectés des journaux et sauvegardés, à condition qu'ils soient disponibles pour être restaurés si nécessaire pour une opération de reprise.

Le concept de gestion de journal est illustré dans la [Figure 35](#), à la page 291.

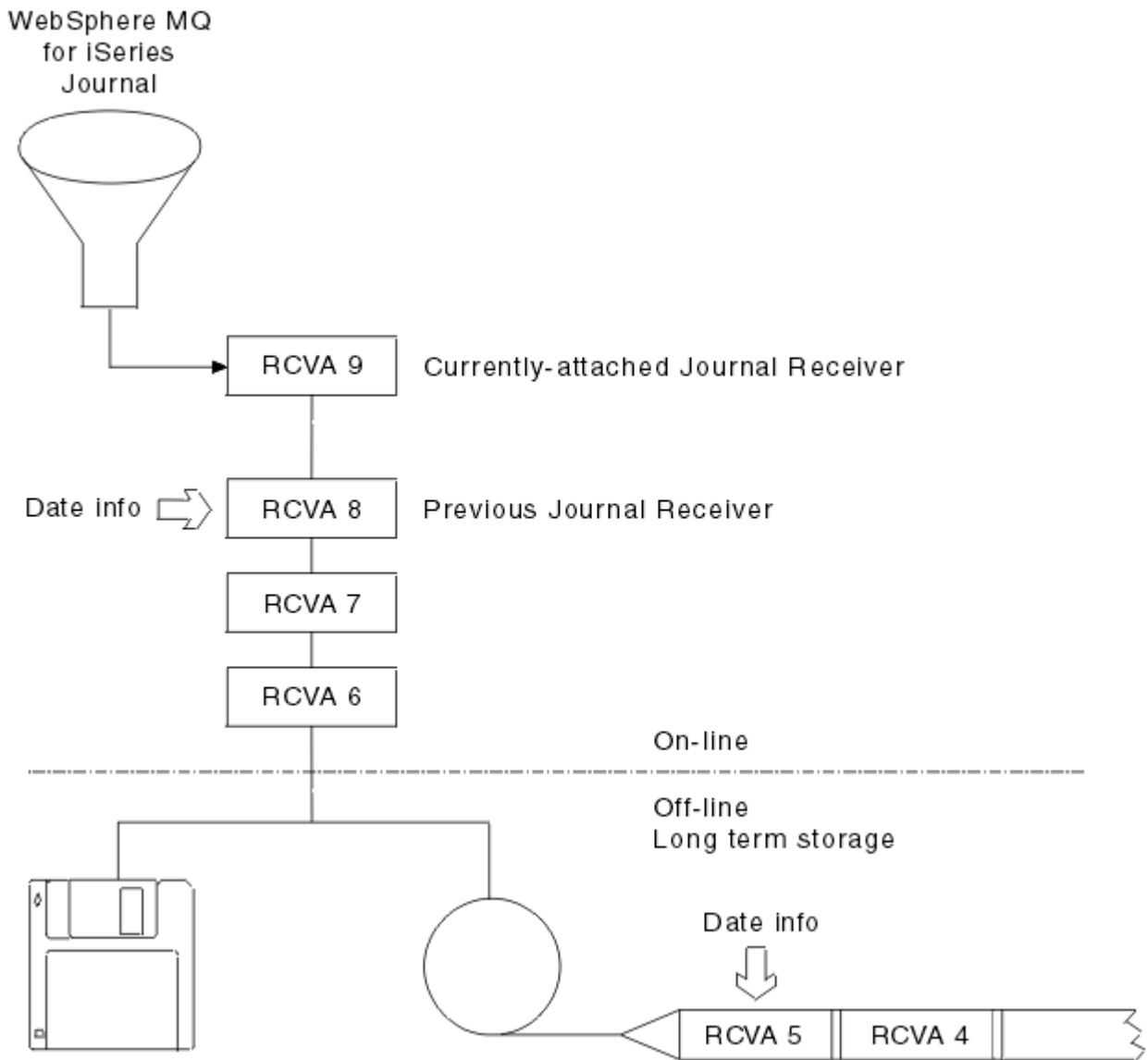


Figure 35. Journalisation sur IBM i

Il est important de savoir jusqu'où se trouvent les journaux IBM MQ , afin de déterminer quand un récepteur de journal qui a été sauvegardé peut être supprimé de la bibliothèque du gestionnaire de files d'attente et quand la sauvegarde elle-même peut être supprimée.

IBM MQ émet deux messages dans la file d'attente de messages du gestionnaire de files d'attente (QMOMMSG dans la bibliothèque du gestionnaire de files d'attente) pour vous aider à déterminer cette heure. Ces messages sont émis lorsqu'il démarre, lorsqu'il modifie un récepteur de journal local et que vous utilisez RCDMQIMG pour forcer un point de contrôle. Les deux messages sont les suivants:

AMQ7460

Point de reprise au démarrage. Ce message définit la date et l'heure de l'entrée de démarrage à partir de laquelle IBM MQ réexécute le journal en cas de reprise au démarrage. Si le récepteur de journal contenant cet enregistrement est disponible dans les bibliothèques IBM MQ , ce message contient également le nom du récepteur de journal contenant l'enregistrement.

AMQ7462

Poste de reprise de support le plus ancien. Ce message définit la date et l'heure de l'entrée la plus ancienne à utiliser pour recréer un objet à partir de son image de support.

Le récepteur de journal identifié est le plus ancien requis. Les autres récepteurs de journal IBM MQ ayant des dates de création plus anciennes ne sont plus nécessaires. Si seules des étoiles sont

affichées, vous devez restaurer les sauvegardes à partir de la date indiquée pour déterminer le récepteur de journal le plus ancien.

Lorsque ces messages sont consignés, IBM MQ écrit également un objet d'espace utilisateur dans la bibliothèque du gestionnaire de files d'attente qui contient une seule entrée: le nom du récepteur de journal le plus ancien qui doit être conservé sur le système. Cet espace utilisateur est appelé AMQJRNINF et les données sont écrites au format suivant:

```
JJJJJJJJJLLLLLLLLLLLLYYYYMMDDHHMMSSmmm
```

où :

JJJJJJJJJ

Est le nom de récepteur le plus ancien dont IBM MQ a toujours besoin.

LLLLLLLLLL

Nom de la bibliothèque du récepteur de journal.

YYYY

Année de l'entrée de journal la plus ancienne dont IBM MQ a besoin.

MM

Mois de l'entrée de journal la plus ancienne dont IBM MQ a besoin.

DD

Jour de l'entrée de journal la plus ancienne dont IBM MQ a besoin.

HH

Heure de l'entrée de journal la plus ancienne dont IBM MQ a besoin.

SS

Nombre de secondes de l'entrée de journal la plus ancienne dont IBM MQ a besoin.

mmm

Nombre de millisecondes de l'entrée de journal la plus ancienne dont IBM MQ a besoin.

Lorsque le récepteur de journal le plus ancien a été supprimé du système, cet espace utilisateur contient des astérisques (*) pour le nom du récepteur de journal.

Remarque : L'exécution périodique de RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) DSPJRNDTA(*YES) permet d'économiser du temps de démarrage pour IBM MQ et de réduire le nombre de récepteurs de journal locaux dont vous avez besoin pour la sauvegarde et la restauration en vue de la reprise.

IBM MQ for IBM i ne fait référence aux récepteurs de journal que s'il effectue un processus de reprise soit pour le démarrage, soit pour la recréation d'un objet. S'il détecte qu'un journal dont il a besoin n'est pas présent, il émet le message AMQ7432 dans la file d'attente de messages du gestionnaire de files d'attente (QMQMMSG), en indiquant l'heure et la date de l'entrée de journal dont il a besoin pour terminer le processus de reprise.

Si tel est le cas, restaurez tous les récepteurs de journal qui ont été déconnectés après cette date à partir de la sauvegarde, afin de permettre la réussite du processus de reprise.

Conservez le récepteur de journal qui contient le poste de démarrage, ainsi que les récepteurs de journal suivants, disponibles dans la bibliothèque du gestionnaire de files d'attente.

Conservez le récepteur de journal contenant le plus ancien Media Recovery Entry, ainsi que tous les récepteurs de journal suivants, disponibles à tout moment et présents dans la bibliothèque du gestionnaire de files d'attente ou sauvegardés.

Lorsque vous forcez un point de contrôle:

- Si le récepteur de journal nommé dans AMQ7460 n'est pas avancé, cela indique qu'une unité de travail incomplète doit être validée ou annulée.
- Si le récepteur de journal nommé dans AMQ7462 n'est pas avancé, cela indique qu'il existe un ou plusieurs objets endommagés.

Restoration d'un gestionnaire de files d'attente complet (données et journaux) sous IBM i

Utilisez ces informations pour restaurer un ou plusieurs gestionnaires de files d'attente à partir d'une sauvegarde ou d'une machine distante.

Si vous devez récupérer un ou plusieurs gestionnaires de files d'attente IBM MQ à partir d'une sauvegarde, procédez comme suit.

1. Mettez au repos les gestionnaires de files d'attente IBM MQ .
2. Localisez votre dernier groupe de sauvegarde, constitué de votre sauvegarde intégrale la plus récente et des récepteurs de journal sauvegardés par la suite.
3. Effectuez une opération RSTLIB, à partir de la sauvegarde intégrale, pour restaurer les bibliothèques de données IBM MQ à leur état au moment de la sauvegarde intégrale, en exécutant les commandes suivantes:

```
RSTLIB LIB(QMQLIB1) .....
RSTLIB LIB(QMQLIB2) .....
```

Si un récepteur de journal a été partiellement sauvegardé dans une sauvegarde de journal et entièrement sauvegardé dans une sauvegarde ultérieure, restaurez uniquement la sauvegarde complète. Restaurez les journaux individuellement, dans l'ordre chronologique.

4. Effectuez une opération RST pour la restauration des répertoires IFS IBM MQ sur le système de fichiers IFS, à l'aide de la commande suivante:

```
RST DEV(...) OBJ((' /QIBM/UserData/mqm/qmgrs/testqm')) ...
```

5. Démarrez le gestionnaire de files d'attente de messages. Cette commande réexécute tous les enregistrements de journal écrits depuis la sauvegarde intégrale et restaure tous les objets IBM MQ à l'état cohérent au moment de la sauvegarde du journal.

Si vous souhaitez restaurer un gestionnaire de files d'attente complet sur une machine différente, utilisez la procédure suivante pour tout restaurer à partir de la bibliothèque du gestionnaire de files d'attente. (Nous utilisons TEST comme exemple de nom de gestionnaire de files d'attente.)

1. CRTMQM TEST
2. DLTLIB LIB(QMTEST)
3. RSTLIB SAVLIB(QMTEST) DEV(*SAVF) SAVF(QMGRLIBSAV)
4. Supprimez les fichiers IFS suivants:

```
/QIBM/UserData/mqm/qmgrs/TEST/QMQMCHKPT
/QIBM/UserData/mqm/qmgrs/TEST/qmanager/QMQMOBJCAT
/QIBM/UserData/mqm/qmgrs/TEST/qmanager/QMANAGER
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.AUTH.DATA.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CHANNEL.INITQ/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.COMMAND.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.REPOSITORY.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.CLUSTER.TRANSMIT.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.PENDING.DATA.QUEUE/q
/QIBM/UserData/mqm/qmgrs/TEST/queues/SYSTEM.ADMIN.COMMAND.QUEUE/q
```

5. STRMQM TEST
6. RCRMQMOBJ OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(TEST)

Restauration de récepteurs de journal pour un gestionnaire de files d'attente particulier sous IBM i

Utilisez ces informations pour comprendre les différentes manières de restaurer les récepteurs de journal.

L'action la plus courante consiste à restaurer un récepteur de journal sauvegardé dans une bibliothèque de gestionnaire de files d'attente, si un récepteur qui a été supprimé est à nouveau nécessaire pour une fonction de reprise ultérieure.

Il s'agit d'une tâche simple qui nécessite la restauration des récepteurs de journal à l'aide de la commande IBM i RSTOBJ standard:

```
RSTOBJ OBJ(QMQMDATA/AMQA000005) OBJTYPE(*JRNRCV) .....
```

Il peut être nécessaire de restaurer une série de récepteurs de journal plutôt qu'un seul. Par exemple, AMQA000007 est le récepteur le plus ancien dans les bibliothèques IBM MQ et AMQA000005 et AMQA000006 doivent être restaurés.

Dans ce cas, restaurez les récepteurs individuellement dans l'ordre chronologique inverse. Ce n'est pas toujours nécessaire, mais c'est une bonne pratique. Dans les situations graves, vous devrez peut-être utiliser la IBM i commande WRKJRNA pour associer les récepteurs de journal restaurés au journal.

Lors de la restauration de journaux, le système crée automatiquement un récepteur de journal attaché avec un nouveau nom dans la séquence de récepteurs de journal. Toutefois, le nouveau nom généré peut être identique à celui d'un récepteur de journal que vous devez restaurer. Une intervention manuelle est nécessaire pour résoudre ce problème ; pour créer un nouveau récepteur de journal de nom dans l'ordre et un nouveau journal avant de restaurer le récepteur de journal.

Par exemple, prenez en compte l'incident lié au journal sauvegardé AMQAJRN et aux récepteurs de journal suivants:

- AMQA000000
- AMQA100000
- AMQA200000
- AMQA300000
- AMQA400000
- AMQA500000
- AMQA600000
- AMQA700000
- AMQA800000
- AMQA900000

Lors de la restauration du journal AMQAJRN dans une bibliothèque de gestionnaire de files d'attente, le système crée automatiquement le récepteur de journal AMQA000000. Ce récepteur généré automatiquement entre en conflit avec l'un des récepteurs de journal existants (AMQA000000) que vous souhaitez restaurer et que vous ne pouvez pas restaurer.

La solution est la suivante :

1. Créez manuellement le récepteur de journal suivant (voir [«Journaux de gestionnaire de files d'attente sous IBM i»](#), à la page 285):

```
CRTJRNRCV JRNRCV(QMQRLIB/AMQA900001) THRESHOLD(XXXXX)
```

2. Créez manuellement le journal avec le récepteur de journal:

```
CRTJRN JRN(QMGLIB/AMQAJRN) MNGRCV(*SYSTEM) +
JRNRVC(QMGLIB/AMQA9000001) MSGQ(QMGLIB/AMQAJRNMSG)
```

3. Restaurez les récepteurs de journal locaux AMQA000000 sur AMQA900000.

IBM i

Gestionnaires de files d'attente multi-instances sous IBM i

Les gestionnaires de files d'attente multi-instances permettent d'améliorer la disponibilité du système en basculant automatiquement vers un serveur de secours en cas de défaillance du serveur actif. Les serveurs actifs et de secours hébergent plusieurs instances d'un même gestionnaire de files d'attente et partagent des données identiques le concernant. Si l'instance active échoue, vous devez transférer son journal vers la base de données de secours qui prend le relais afin que le gestionnaire de files d'attente puisse régénérer ses files d'attente.

Configurez les systèmes IBM i sur lesquels vous exécutez des gestionnaires de files d'attente multi-instance de sorte que, si l'instance de gestionnaire de files d'attente active échoue, le journal qu'elle utilise soit disponible pour l'instance de secours qui prend le relais. Vous pouvez concevoir vos propres tâches de configuration et d'administration pour rendre le journal de l'instance active disponible à l'instance qui prend le relais. Si vous ne souhaitez pas perdre de messages, votre conception doit vous assurer que le journal de secours est cohérent avec le journal actif au point de défaillance. Vous pouvez adapter votre conception à partir de l'une des deux configurations décrites avec des exemples dans les rubriques suivantes qui assurent la cohérence.

1. Mettez en miroir le journal du système qui exécute l'instance de gestionnaire de files d'attente active vers les systèmes qui exécutent des instances de secours.
2. Placez le journal dans un pool de mémoire secondaire indépendant (IASP) qui est transférable du système exécutant l'instance active vers une instance de secours.

La première solution ne nécessite aucun matériel ou logiciel supplémentaire car elle utilise des ASP de base. La deuxième solution nécessite des ASP indépendants commutables qui ont besoin d'une prise en charge de la mise en cluster IBM i disponible sous la forme d'un IBM i produit de licence 5761-SS1 à prix séparé.

IBM i

Fiabilité et disponibilité sous IBM i

Les gestionnaires de files d'attente multi-instance visent à améliorer la disponibilité des applications. Les contraintes technologiques et physiques signifient que vous avez besoin de solutions différentes pour répondre aux exigences de la reprise après incident, de la sauvegarde des gestionnaires de files d'attente et de l'exploitation continue.

Lors de la configuration de la fiabilité et de la disponibilité, vous échangez un grand nombre de facteurs, ce qui se traduit par quatre points de conception distincts:

Reprise après incident

Optimisé pour la reprise après un sinistre majeur qui détruit tous vos actifs locaux.

La reprise après incident sur IBM i est souvent basée sur la protection géographique par disque miroir de l'ASP indépendant.

Sauvegarde

Optimisé pour la récupération après une panne localisée, généralement une erreur humaine ou un problème technique imprévu.

IBM MQ fournit des gestionnaires de files d'attente de sauvegarde pour la sauvegarde périodique des gestionnaires de files d'attente. Vous pouvez également utiliser la réplication asynchrone des journaux de gestionnaire de files d'attente pour améliorer la mise à jour de la sauvegarde.

Disponibilité

Optimisé pour la restauration rapide des opérations donnant l'apparence d'un service quasi ininterrompu suite à des défaillances techniques prévisibles telles qu'une panne de serveur ou de disque.

La récupération est généralement mesurée en minutes, la détection prenant parfois plus de temps que le processus de récupération. Un gestionnaire de files d'attente multi-instance vous aide à configurer la *disponibilité*.

Fonctionnement en continu

Optimisé pour fournir un service ininterrompu.

Les solutions de fonctionnement continu doivent résoudre le problème de détection, et impliquent presque toujours de soumettre le même travail à travers plus d'un système et soit en utilisant le premier résultat, soit si l'exactitude est une considération majeure, en comparant au moins deux résultats.

Un gestionnaire de files d'attente multi-instance vous aide à configurer la *disponibilité*. Une instance du gestionnaire de files d'attente est active à la fois. Le basculement vers une instance de secours prend entre un peu plus de dix secondes et une quinzaine de minutes ou plus, selon la manière dont le système est configuré, chargé et réglé.

Un gestionnaire de files d'attente multi-instance peut donner l'apparence d'un service presque ininterrompu s'il est utilisé avec des IBM MQ MQI clients reconnectables, qui peuvent continuer le traitement sans que le programme d'application ait nécessairement connaissance d'une indisponibilité du gestionnaire de files d'attente ; voir la rubrique [Reconnexion client automatisée](#).



Composants d'une solution à haute disponibilité sur IBM i

Construisez une solution à haute disponibilité à l'aide de gestionnaires de files d'attente multi-instance en fournissant un stockage en réseau robuste pour les données de gestionnaire de files d'attente, une réplication de journal ou un stockage IASP robuste pour les journaux de gestionnaire de files d'attente, et en utilisant des clients reconnectables, d'applications configurées en tant que services de gestionnaire de files d'attente redémarrables.

Un gestionnaire de files d'attente multi-instance réagit à la détection de l'échec du gestionnaire de files d'attente en reprenant le démarrage d'une autre instance de gestionnaire de files d'attente sur un autre serveur. Pour terminer son démarrage, l'instance doit accéder aux données du gestionnaire de files d'attente partagées dans le stockage en réseau et à sa copie du journal du gestionnaire de files d'attente local.

Pour créer une solution à haute disponibilité, vous devez gérer la disponibilité des données du gestionnaire de files d'attente, la mise à jour du journal du gestionnaire de files d'attente local et générer des applications client reconnectables ou déployer vos applications en tant que services de gestionnaire de files d'attente pour redémarrer automatiquement lorsque le gestionnaire de files d'attente reprend. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java.

Données de gestionnaire de files d'attente

Placez les données du gestionnaire de files d'attente dans un stockage en réseau partagé, hautement disponible et fiable, éventuellement à l'aide de disques RAID de niveau 1 ou supérieur. Le système de fichiers doit répondre aux exigences d'un système de fichiers partagé pour les gestionnaires de files d'attente multi-instance. Pour plus d'informations sur les exigences pour les systèmes de fichiers partagés, voir [Exigences pour les systèmes de fichiers partagés](#). Network File System version 4 (NFS4) est un protocole qui répond à ces exigences.

Journaux du gestionnaire de files d'attente

Vous devez également configurer les journaux IBM i utilisés par les instances de gestionnaire de files d'attente afin que l'instance de secours puisse restaurer les données de son gestionnaire de files d'attente à un état cohérent. Pour un service ininterrompu, cela signifie que vous devez restaurer les journaux à leur état lorsque l'instance active a échoué. Contrairement aux solutions de sauvegarde ou de reprise après incident, la restauration des journaux à un point de contrôle antérieur n'est pas suffisante.

Vous ne pouvez pas partager physiquement des journaux entre plusieurs systèmes IBM i sur le stockage en réseau. Pour restaurer les journaux de gestionnaire de files d'attente à l'état cohérent au point de défaillance, vous devez soit transférer le journal physique qui était local à l'instance de gestionnaire de

files d'attente active au moment de la défaillance à la nouvelle instance qui a été activée, soit conserver des miroirs du journal sur les instances de secours en cours d'exécution. Le journal en miroir est une réplique de journal éloigné qui a été maintenue exactement en synchronisation avec le journal local appartenant à l'instance ayant échoué.

Trois configurations sont des points de départ pour la conception de la gestion des journaux d'un gestionnaire de files d'attente multi-instance.


1. Utilisation de la réplication de journal synchronisée (protection par disque miroir) à partir de l'ASP de l'instance active vers les ASP des instances de secours.
2. Transfert d'un ASP indépendant que vous avez configuré pour stocker le journal du gestionnaire de files d'attente de l'instance active vers l'instance de secours qui prend le relais en tant qu'instance active.
3. Utilisation de miroirs d'ASP indépendant secondaires synchronisés.

Pour plus d'informations sur l'insertion de données de gestionnaire de files d'attente dans un iASP, voir les options [ASP](#) dans la commande IBM MQ IBM i CRTMQM.

Voir aussi [Haute disponibilité](#) dans IBM Documentation.

Applications

Pour générer un client afin de se reconnecter automatiquement au gestionnaire de files d'attente lors de la reprise du gestionnaire de files d'attente de secours, connectez votre application au gestionnaire de files d'attente à l'aide de MQCONNX et spécifiez MQCNO_RECONNECT_Q_MGR dans la zone Options de **MQCNO**. Voir [High availability sample programs](#) pour trois exemples de programmes utilisant des clients reconnectables et [Application recovery](#) pour plus d'informations sur la conception d'applications client pour la reprise.

 *Création d'un partage de réseau pour les données de gestionnaire de files d'attente à l'aide de NetServer sous IBM i*

Créez un partage de réseau sur un serveur IBM i pour stocker les données du gestionnaire de files d'attente. Configurez des connexions à partir de deux serveurs, qui vont héberger des instances de gestionnaire de files d'attente, pour accéder au partage de réseau.

Avant de commencer

- Vous avez besoin de trois serveurs IBM i pour cette tâche. Le partage de réseau est défini sur l'un des serveurs, GAMMA. Les deux autres serveurs, ALPHA et BETA, doivent se connecter à GAMMA.
- Installez IBM MQ sur les trois serveurs.
- Installez System i Navigator; voir [System i Navigator](#).

Pourquoi et quand exécuter cette tâche

- Créez le répertoire du gestionnaire de files d'attente sur GAMMA et définissez la propriété et les droits appropriés pour les profils utilisateur QMQM et QMQMADM. Le répertoire et les droits d'accès sont facilement créés en installant IBM MQ sur GAMMA.
- Utilisez System i Navigator pour créer un partage dans le répertoire de données du gestionnaire de files d'attente sur GAMMA.
- Créez des répertoires sur ALPHA et BETA qui pointent vers le partage.

Procédure

1. Sous GAMMA, créez le répertoire pour héberger les données du gestionnaire de files d'attente avec le profil utilisateur QMQM comme propriétaire et QMQMADM comme groupe principal.

Conseil :

Un moyen rapide et fiable de créer le répertoire avec les droits appropriés consiste à installer IBM MQ sur GAMMA.

Par la suite, si vous ne souhaitez pas exécuter IBM MQ sur GAMMA, désinstallez IBM MQ. Après la désinstallation, le répertoire /QIBM/UserData/mqm/qmgrs reste sur GAMMA avec le profil utilisateur QMQM propriétaire et QMQMADM le groupe principal.

La tâche utilise le répertoire /QIBM/UserData/mqm/qmgrs sur GAMMA pour le partage.

2. Démarrez l'assistant System i Navigator **Ajouter une connexion** et connectez-vous au système GAMMA.

- a) Cliquez deux fois sur l'icône **System i Navigator** sur votre bureau Windows .
- b) Cliquez sur **Oui** pour créer une connexion.
- c) Suivez les instructions de l'assistant **Ajouter une connexion** et créez une connexion entre le système IBM i et GAMMA.

La connexion à GAMMA est ajoutée à **Mes connexions**.

3. Ajoutez un nouveau partage de fichiers sur GAMMA.

- a) Dans la fenêtre **System i Navigator** , cliquez sur le dossier File Shares dans My Connections/GAMMA/File Systems.

- b) Dans la fenêtre **Mes tâches** , cliquez sur **Gérer les partages IBM i NetServer**.

Une nouvelle fenêtre, **IBM i NetServer -GAMMA**, s'ouvre sur votre bureau et affiche les objets partagés.

- c) Cliquez avec le bouton droit de la souris sur le dossier Shared Objects > **Fichier** > **Nouveau** > **Fichier**.

Une nouvelle fenêtre, **IBM i NetServer File Share-GAMMA**, s'ouvre.

- d) Attribuez un nom au partage, WMQ par exemple.

- e) Définissez le contrôle d'accès sur Read/Write.

- f) Sélectionnez le **Nom de chemin** en accédant au répertoire /QIBM/UserData/mqm/qmgrs que vous avez créé précédemment, puis cliquez sur **OK**.

La fenêtre **IBM i NetServer File Share-GAMMA** se ferme et WMQ est répertorié dans la fenêtre des objets partagés.

4. Cliquez avec le bouton droit de la souris sur **WMQ** dans la fenêtre des objets partagés. Cliquez sur **Fichier** > **Droits d'accès**.

Une fenêtre s'ouvre, **Qmgrs Permissions-GAMMA**, pour l'objet /QIBM/UserData/mqm/qmgrs.

- a) Vérifiez les droits suivants pour QMQM, s'ils ne sont pas déjà définis:

- Read
- Write
- Execute
- Management
- Existence
- Alter
- Reference

- b) Vérifiez les droits suivants pour QMQMADM, s'ils ne sont pas déjà définis:

- Read
- Write
- Execute
- Reference

- c) Ajoutez d'autres profils utilisateur auxquels vous souhaitez accorder des droits d'accès à /QIBM/UserData/mqm/qmgrs.

Par exemple, vous pouvez accorder au profil utilisateur par défaut (public) Read et Execute les droits d'accès à /QIBM/UserData/mqm/qmgrs.

5. Vérifiez que tous les profils utilisateur autorisés à accéder à /QIBM/UserData/mqm/qmgrs sur GAMMA ont le même mot de passe que sur les serveurs qui accèdent à GAMMA.

En particulier, assurez-vous que les profils utilisateur QMQM sur d'autres serveurs, qui vont accéder au partage, ont le même mot de passe que le profil utilisateur QMQM sur GAMMA.

Conseil : Cliquez sur le dossier My Connections/GAMMA/Users and Groups dans le System i Navigateur Navigator pour définir les mots de passe. Vous pouvez également utiliser les commandes **CHFUSRPRF** et **CHGPWD**.

Résultats

Vérifiez que vous pouvez accéder à GAMMA à partir d'autres serveurs à l'aide du partage. Si vous effectuez les autres tâches, vérifiez que vous pouvez accéder à GAMMA à partir de ALPHA et de BETA en utilisant le chemin /QNTC/GAMMA/WMQ. Si le répertoire /QNTC/GAMMA n'existe pas sur ALPHA ou BETA, vous devez le créer. Selon le domaine NetServer, vous devrez peut-être effectuer un IPL ALPHA ou BETA avant de créer le répertoire.

```
CRTDIR DIR('/QNTC/GAMMA')
```

Une fois que vous avez vérifié que vous avez accès à /QNTC/GAMMA/WMQ à partir de ALPHA ou de BETA, en émettant la commande, `CRTMQM MQMNAME('QM1') MQMDIRP('/QNTC/GAMMA/WMQ')` crée /QIBM/UserData/mqm/qmgrs/QM1 sur GAMMA.

Que faire ensuite

Créez un gestionnaire de files d'attente multi-instance en suivant les étapes de l'une des tâches, [«Création d'un gestionnaire de files d'attente multi-instance à l'aide de la mise en miroir du journal et de NetServer sous IBM i»](#), à la page 311 ou [«Conversion d'un gestionnaire de files d'attente à instance unique en gestionnaire de files d'attente multi-instance à l'aide de NetServer et de la mise en miroir du journal sous IBM i»](#), à la page 315.

IBM i

Performances de reprise en ligne sous IBM i

Le temps nécessaire pour détecter une instance de gestionnaire de files d'attente a échoué, puis pour reprendre le traitement sur une instance de secours peut varier entre des dizaines de secondes et quinze minutes ou plus selon la configuration. Les performances doivent être une considération majeure lors de la conception et du test d'une solution à haute disponibilité.

Il existe des avantages et des inconvénients à prendre en compte lors de la configuration d'un gestionnaire de files d'attente multi-instance pour utiliser la réplication de journal ou un ASP indépendant. La mise en miroir nécessite que le gestionnaire de files d'attente écrive de manière synchrone dans un journal éloigné. D'un point de vue matériel, cela n'a pas besoin d'affecter les performances, mais du point de vue logiciel, il y a une plus grande longueur de chemin d'accès à l'écriture dans un journal éloigné que dans un journal local, ce qui peut réduire les performances d'un gestionnaire de files d'attente en cours d'exécution dans une certaine mesure. Toutefois, lorsque le gestionnaire de files d'attente de secours prend le relais, le délai de synchronisation de son journal local à partir du journal éloigné géré par l'instance active avant son échec est généralement faible par rapport au temps nécessaire à IBM i pour détecter et transférer l'ASP indépendant sur le serveur exécutant l'instance de secours du gestionnaire de files d'attente. Les temps de transfert de l'ASP indépendant peuvent atteindre jusqu'à dix à quinze minutes au lieu d'être terminés en secondes. Le temps de transfert de l'ASP indépendant dépend du nombre d'objets qui doivent être *mis en fonction* lorsque l'ASP indépendant est transféré sur le système de secours et de la taille des chemins d'accès, ou des index, qui doivent être fusionnés.

Lorsque le gestionnaire de files d'attente de secours prend le relais, le délai de synchronisation de son journal local à partir du journal éloigné géré par l'instance active avant l'échec est généralement faible par rapport au temps nécessaire à IBM i pour détecter et transférer l'ASP indépendant sur le

serveur exécutant l'instance de secours du gestionnaire de files d'attente. Les temps de transfert de l'ASP indépendant peuvent aller jusqu'à dix à quinze minutes au lieu d'être terminés en quelques secondes. La durée de transfert de l'ASP indépendant dépend du nombre d'objets qui doivent être *mis en fonction* lorsque l'ASP indépendant est transféré sur le système de secours et de la taille des chemins d'accès, ou des index, qui doivent être fusionnés.

Toutefois, le transfert du journal n'est pas le seul facteur qui influence le temps nécessaire à la reprise complète de l'instance de secours. Vous devez également prendre en compte le temps nécessaire au système de fichiers du réseau pour libérer le verrou sur les données du gestionnaire de files d'attente qui indiquent à l'instance de secours de tenter de poursuivre son démarrage, ainsi que le temps nécessaire à la reprise des files d'attente à partir du journal afin que l'instance puisse recommencer à traiter les messages. Ces autres sources de retard s'ajoutent au temps nécessaire au démarrage d'une instance de secours. La durée totale du basculement comprend les composants suivants:

Heure de détection des incidents

Temps nécessaire à NFS pour libérer le verrou sur les données du gestionnaire de files d'attente et à l'instance de secours pour poursuivre son processus de démarrage.

Durée de transfert

Dans le cas d'un cluster à haute disponibilité, le temps nécessaire à IBM i pour transférer l'ASP indépendant du système hébergeant l'instance active vers l'instance de secours, et dans le cas de la réplication de journal, le temps nécessaire à la mise à jour du journal local sur la base de secours avec les données de la réplique distante.

Heure de redémarrage

Temps nécessaire à l'instance de gestionnaire de files d'attente nouvellement active pour régénérer ses files d'attente à partir du dernier point de contrôle dans son journal restauré et pour reprendre le traitement des messages.

Remarque :

Si l'instance de secours qui a pris le relais est configurée pour être répliquée de manière synchrone sur l'instance active précédente, le démarrage peut être retardé. Il se peut que la nouvelle instance activée ne puisse pas être répliquée dans son journal distant, si le journal distant se trouve sur le serveur qui a hébergé l'instance active précédente et que le serveur ait échoué.

Le délai d'attente par défaut d'une réponse synchrone est d'une minute. Vous pouvez configurer le délai maximal avant l'expiration de la réplication. Vous pouvez également configurer des instances de secours pour commencer à utiliser la réplication asynchrone sur l'instance active qui a échoué. Vous basculez ensuite vers la réplication synchrone, lorsque l'instance ayant échoué s'exécute à nouveau sur la base de données de secours. Il en va de même pour l'utilisation de miroirs ASP indépendants synchrones.

Vous pouvez effectuer des mesures de référence distinctes pour ces composants pour vous aider à évaluer la durée globale de la reprise en ligne et à prendre en compte dans votre décision l'approche de configuration à utiliser. Pour prendre la meilleure décision en matière de configuration, vous devez également tenir compte de la manière dont d'autres applications sur le même serveur basculeront et de la présence ou non de processus de sauvegarde ou de reprise après incident qui utilisent déjà l'ASP indépendant.

Les temps de transfert ASP indépendant peuvent être raccourcis en optimisant la configuration de votre grappe:

1. Les profils utilisateur des systèmes du cluster doivent avoir le même ID groupe et le même ID utilisateur pour que le processus de mise en fonction n'ait pas besoin de modifier les ID utilisateur et les ID groupe.
2. Réduisez le nombre d'objets de base de données dans le système et les pools de stockage sur disque utilisateur de base, car ils doivent être fusionnés pour créer la table de références croisées pour le groupe de pools de stockage sur disque.
3. D'autres conseils sur les performances sont disponibles dans le document IBM Redbook, *Implémentation PowerHA for IBM i, SG24-7405*.

Une configuration utilisant des ASP de base, la mise en miroir du journal et une configuration de petite taille doit basculer de l'ordre de plusieurs dizaines de secondes.

IBM i *Présentation de la combinaison des fonctions de mise en cluster IBM i avec la mise en cluster IBM MQ*

L'exécution de IBM MQ sous IBM i et l'exploitation des fonctions de mise en cluster IBM i peuvent fournir une solution à haute disponibilité plus complète que l'utilisation de la mise en cluster IBM MQ uniquement.

Pour disposer de cette fonctionnalité, vous devez configurer:

1. Clusters sur votre machine IBM i ; voir [«Clusters IBM i»](#), à la page 301
2. Un pool de mémoire secondaire indépendant (IASP) dans lequel vous déplacez le gestionnaire de files d'attente ; voir [«Pools de mémoire secondaire indépendants \(IASP\)»](#), à la page 301
3. Un groupe de ressources en grappe (CRG) ; voir [«Groupes de ressources en grappe d'unité»](#), à la page 301, dans lequel vous définissez:
 - Domaine de reprise
 - ASP indépendant
 - Programme d'exit ; voir [«Programme d'exit CRG d'unité»](#), à la page 302

Clusters IBM i

Un cluster IBM i est une collection d'instances, c'est-à-dire d'ordinateurs ou de partitions IBM i , qui sont liées logiquement entre elles.

Le but de ce regroupement est de permettre la sauvegarde de chaque instance, en éliminant un point de défaillance unique et en augmentant la résilience des applications et des données. Avec une grappe créée, les différents types de groupe de ressources en grappe peuvent être configurés pour gérer les applications, les données et les unités de la grappe.

Pour plus d'informations, voir [Création d'une grappe](#) et la commande [CRTCLU \(Créer une grappe\)](#) .

Pools de mémoire secondaire indépendants (IASP)

Un ASP indépendant est un type d'ASP utilisateur qui sert d'extension de l'espace de stockage à un niveau. C'est une pièce de stockage qui, en raison de son indépendance par rapport à la mémoire du système, peut être facilement manipulée sans avoir à effectuer un IPL du système.

Un ASP indépendant peut être facilement basculé vers une autre instance de système d'exploitation ou répliqué vers un ASP indépendant cible sur une autre instance de système d'exploitation. Deux méthodes peuvent être utilisées pour permuter un ASP indépendant entre des instances:

- La première méthode nécessite que tous les ordinateurs de la grappe, et la tour de disque commutable contenant l'ASP indépendant, soient connectés à l'aide d'une boucle HSL (High Speed Link).
- La deuxième méthode requiert que les instances de système d'exploitation soient des partitions sur le même ordinateur IBM i où les processeurs d'entrée-sortie (IOPS) peuvent être commutés entre les partitions. Aucun matériel spécial n'est nécessaire pour pouvoir répliquer un ASP indépendant. La réplication est effectuée à l'aide de TCP/IP sur le réseau.

Pour plus d'informations, voir la commande [Configuration d'ASP d'unité \(CFGDEVASP\)](#) .

Groupes de ressources en grappe d'unité

Il existe plusieurs types de groupes de ressources en grappe (CRG). Pour plus d'informations sur les différents types de groupes de ressources en grappe disponibles, voir [Groupe de ressources en grappe](#).

Cette rubrique se concentre sur un groupe de ressources en grappe d'unité. Un groupe de ressources en grappe d'unité:

- Décrit et gère les ressources d'unité, telles que les pools de mémoire secondaire indépendants (IASP).
- Définit le domaine de reprise des noeuds de grappe
- Affecte un périphérique et
- Affecte le programme d'exit qui traitera les événements de cluster.

Le domaine de reprise indique quel noeud de grappe sera considéré comme le noeud principal. Les autres noeuds sont considérés comme des sauvegardes. Les noeuds de sauvegarde sont également classés dans le domaine de reprise, en spécifiant le noeud qui est la première sauvegarde, la deuxième sauvegarde, etc., en fonction du nombre de noeuds présents dans le domaine de reprise.

En cas de défaillance d'un noeud principal, le programme d'exit est exécuté sur tous les noeuds du domaine de reprise. Le programme d'exit exécuté sur la première sauvegarde peut alors effectuer les initialisations nécessaires pour faire de ce noeud le nouveau noeud principal.

Pour plus d'informations, voir [Création de groupes de ressources en grappe d'unité](#) et la commande [Créer un groupe de ressources en grappe \(CRTCRG\)](#).

Programme d'exit CRG d'unité

Le service de ressources en grappe du système d'exploitation appelle un programme d'exit de groupe de ressources en grappe d'unité lorsqu'un événement se produit dans l'un des noeuds définis par le domaine de reprise ; par exemple, un événement de basculement ou de commutation.

Un événement de reprise en ligne se produit lorsque le noeud principal de la grappe est défaillant et que les groupes de ressources en grappe sont commutés avec toutes les ressources qu'ils gèrent, et un événement de commutation se produit lorsqu'un groupe de ressources en grappe spécifique est basculé manuellement du noeud principal vers le noeud de secours.

Dans les deux cas, le programme d'exit est chargé d'initialiser et de démarrer tous les programmes qui étaient en cours d'exécution sur le noeud principal précédent, ce qui convertit le premier noeud de secours en nouveau noeud principal.

Par exemple, avec IBM MQ, le programme d'exit doit être en charge du démarrage du sous-système IBM MQ (QMQM) et des gestionnaires de files d'attente. Les gestionnaires de files d'attente doivent être configurés pour démarrer automatiquement les programmes d'écoute et les services, tels que les moniteurs de déclenchement.

Configuration d'ASP indépendant commutable

IBM MQ peut être configuré pour tirer parti des capacités de mise en cluster de IBM i. Pour ce faire :

1. Création d'un cluster IBM i entre les systèmes de centre de données
2. Déplacez le gestionnaire de files d'attente vers un ASP indépendant.

«[Déplacement ou suppression d'un gestionnaire de files d'attente vers ou depuis un pool de mémoire secondaire indépendant](#)», à la page 303 contient un exemple de code pour vous aider à effectuer cette opération.

3. Vous devez créer un groupe de ressources en grappe définissant le domaine de reprise, l'ASP indépendant et le programme d'exit.

«[Configuration d'un groupe de ressources en grappe d'unité](#)», à la page 303 contient un exemple de code pour vous aider à effectuer cette opération.

Concepts associés

«[ASP indépendants et haute disponibilité](#)», à la page 324

Les ASP indépendants permettent de déplacer des applications et des données entre les serveurs. La flexibilité des ASP indépendants signifie qu'ils constituent la base de certaines solutions à haute disponibilité IBM i . Lorsque vous envisagez d'utiliser un ASP ou un ASP indépendant pour le journal du gestionnaire de files d'attente, vous devez prendre en compte une autre configuration à haute disponibilité basée sur des ASP indépendants.

Exemple de programme permettant de configurer un groupe de ressources en grappe d'unité.

Pourquoi et quand exécuter cette tâche

Dans l'exemple suivant, notez que:

- [PRIMARY SITE NAME] et [BACKUP SITE NAME] peuvent être deux chaînes distinctes de huit caractères ou moins.
- [PRIMARY IP] et [BACKUP IP] sont les adresses IP à utiliser pour la mise en miroir.

Procédure

1. Identifiez le nom du cluster.
2. Identifiez le nom du programme d'exit du groupe de ressources en grappe et la bibliothèque.
3. Déterminez le nom du noeud principal et des noeuds de sauvegarde à définir par ce groupe de ressources en grappe.
4. Identifiez l'ASP indépendant à gérer par ce groupe de ressources en grappe et assurez-vous qu'il a été créé sous le noeud principal.
5. Créez une description d'unité dans les noeuds de sauvegarde à l'aide de la commande suivante:

```
CRTDEVASP DEVD([IASP NAME]) RSRNAME([IASP NAME])
```

6. Ajoutez l'adresse IP de relais à tous les noeuds à l'aide de la commande suivante:

```
ADDTCPICF INTNETADR(' [TAKEOVER IP]') LIND([LINE DESC])
SUBNETMASK(' [SUBNET MASK]') AUTOSTART(*NO)
```

7. Démarrez l'adresse IP de relais uniquement sur le noeud principal à l'aide de la commande suivante:

```
STRTCPICF INTNETADR(' [TAKEOVER IP]')
```

8. Facultatif : Si votre ASP indépendant est commutable, appelez la commande suivante:

```
CRTCRCG CLUSTER([CLUSTER NAME]) CRG([CRG NAME]) CRGTYPE(*DEV) EXITPGM([EXIT LIB]/[EXIT
NAME])
USRPRF([EXIT PROFILE]) RCYDMN(([PRIMARY NODE] *PRIMARY) ([BACKUP NAME] *BACKUP))
EXITPGMFMT(EXTP0200) CFGOBJ([IASP NAME] *DEV *ONLINE '[TAKEOVER IP]')
```

9. Facultatif : Si votre ASP indépendant doit être protégé par disque miroir, appelez la commande suivante:

```
CRTCRCG CLUSTER([CLUSTER NAME]) CRG([CRG NAME]) CRGTYPE(*DEV) EXITPGM([EXIT LIB]/[EXIT NAME])
USRPRF([EXIT PROFILE]) RCYDMN(([PRIMARY NODE] *PRIMARY *LAST [PRIMARY SITE NAME] ('[PRIMARY
IP]'))
[BACKUP NAME] *BACKUP *LAST [BACKUP SITE NAME] ('[BACKUP IP]')) EXITPGMFMT(EXTP0200)
CFGOBJ([IASP NAME] *DEV *ONLINE '[TAKEOVER IP]'))
```

Exemple de programme permettant de déplacer un gestionnaire de files d'attente vers un pool de mémoire secondaire indépendant (IASP) et de supprimer un gestionnaire de files d'attente d'un IASP.

Pourquoi et quand exécuter cette tâche

Dans l'exemple suivant, notez que:

- [MANAGER NAME] est le nom de votre gestionnaire de files d'attente.
- [IASP NAME] est le nom de votre IASP.

- [MANAGER LIBRARY] est le nom de la bibliothèque de votre gestionnaire de files d'attente.
- [MANAGER DIRECTORY] est le nom de votre répertoire de gestionnaire de files d'attente.

Procédure

1. Identifiez votre noeud principal et vos noeuds de sauvegarde.
2. Effectuez la procédure suivante sur votre noeud principal:
 - a) Vérifiez que votre gestionnaire de files d'attente est arrêté.
 - b) Vérifiez que votre ASP indépendant est `vary on` à l'aide de la commande

```
VRYCFG CFGOBJ([IASP NAME]) CFGTYPE(*DEV) STATUS(*ON)
```

- c) Créez le répertoire des gestionnaires de files d'attente sous l'ASP indépendant.
Il y aura un répertoire sous la racine avec le nom de votre ASP indépendant, à savoir:

```
QSH CMD('mkdir -p /[IASP_NAME]/QIBM/UserData/mqm/qmgrs/')
```

- d) Déplacez les objets IFS de votre gestionnaire dans le répertoire des gestionnaires de files d'attente que vous venez de créer sous l'ASP indépendant à l'aide de la commande suivante:

```
QSH CMD('mv /QIBM/UserData/mqm/qmgrs/[MANAGER NAME]
/[IASP_NAME]/QIBM/UserData/mqm/qmgrs/')
```

- e) Créez un fichier sauvegarde temporaire nommé `MGRLIB` à l'aide de la commande suivante:

```
CRTSAVF QGPL/MGRLIB
```

- f) Sauvegardez votre bibliothèque de gestionnaires de files d'attente dans le fichier sauvegarde `MGRLIB` à l'aide de la commande suivante:

```
SAVLIB LIB([MANAGER LIBRARY]) DEV(*SAVF) SAVF(QGPL/MGRLIB)
```

- g) Supprimez la bibliothèque du gestionnaire de files d'attente à l'aide de la commande suivante et ignorez tous les messages d'interrogation:

```
DLTLIB [MANAGER LIBRARY]
```

- h) Restaurez la bibliothèque de votre gestionnaire de files d'attente dans l'ASP indépendant à l'aide de la commande suivante:

```
RSTLIB SAVLIB([MANAGER LIBRARY]) DEV(*SAVF) SAVF(QGPL/MGRLIB)
RSTASPDEV([IASP NAME])
```

- i) Supprimez le fichier sauvegarde temporaire à l'aide de la commande suivante:

```
DLTF FILE(QGPL/MGRLIB)
```

- j) Créez un lien symbolique vers les objets IFS du gestionnaire de files d'attente sous l'ASP indépendant à l'aide de la commande suivante:

```
ADDLNK OBJ('/[IASP_NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
NEVLNK('/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

- k) Effectuez une connexion à l'ASP indépendant à l'aide de la commande suivante:

```
SETASPGRP [IASP NAME]
```

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

```
STRMQM [MANAGER NAME]
```


3. Exécutez la procédure suivante sur votre ou vos noeuds de sauvegarde:

a) Créez un répertoire de gestionnaire de files d'attente temporaire à l'aide de la commande suivante:

```
QSH CMD('mkdir -p /[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

b) Créez un lien symbolique vers le répertoire temporaire du gestionnaire de files d'attente à l'aide de la commande suivante:

```
ADDLNK OBJ('/[IASP NAME]/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')  
NEWLNK('/QIBM/UserData/mqm/qmgrs/[MANAGER NAME]')
```

c) Supprimez le répertoire temporaire à l'aide de la commande suivante:

```
QSH CMD('rm -r /[IASP NAME]')
```

d) Ajoutez ce qui suit à la fin du fichier /QIBM/UserData/mqm/mqs.ini:

```
QueueManager:  
Name=[MANAGER NAME]  
Prefix=/QIBM/UserData/mqm  
Library=[MANAGER LIBRARY]  
Directory=[MANAGER DIRECTORY]
```

4. Pour supprimer un gestionnaire de files d'attente d'un ASP indépendant, exécutez les commandes suivantes:

a) VRYCFG CFGOBJ ([NOM_FERMOIR]) CFGTYPE (*DEV) STATUS (*ON)

b) SETASPGRP [NOM ASP indépendant]

c) ENDMQM [NOM DU GERE]

d) DLTMQM [NOM DU GERE]

Configuration de journal protégé par disque miroir pour ASP sous IBM i

Configurez un gestionnaire de files d'attente multi-instance robuste à l'aide de la réplication synchrone entre les journaux en miroir.

Une configuration de gestionnaire de files d'attente protégée par disque miroir utilise des journaux créés dans des pools de mémoire secondaire (ASP) de base ou indépendants.

Sous IBM i, les données du gestionnaire de files d'attente sont écrites dans des journaux et dans un système de fichiers. Les journaux contiennent la copie principale des données du gestionnaire de files d'attente. Les journaux sont partagés entre les systèmes à l'aide de la réplication de journal synchrone ou asynchrone. Une combinaison de journaux locaux et éloignés est requise pour redémarrer une instance de gestionnaire de files d'attente. Le redémarrage du gestionnaire de files d'attente lit les enregistrements de journal à partir de la combinaison de journaux locaux et éloignés sur le serveur et des données du gestionnaire de files d'attente sur le système de fichiers réseau partagé. Les données du système de fichiers accélèrent le redémarrage du gestionnaire de files d'attente. Les points de contrôle sont stockés dans le système de fichiers, marquant les points de synchronisation entre le système de fichiers et les journaux. Les enregistrements de journal stockés avant le point de contrôle ne sont pas requis pour les redémarrages classiques du gestionnaire de files d'attente. Toutefois, il se peut que les données du système de fichiers ne soient pas à jour et que les enregistrements de journal après le point de contrôle soient utilisés pour terminer le redémarrage du gestionnaire de files d'attente. Les données des journaux associés à l'instance sont maintenues à jour afin que le redémarrage aboutisse.

Mais même les enregistrements de journal peuvent ne pas être à jour, si le journal distant sur le serveur de secours était répliqué de manière asynchrone et que l'incident s'est produit avant qu'il ne soit synchronisé. Si vous décidez de redémarrer un gestionnaire de files d'attente à l'aide d'un journal éloigné qui n'est pas synchronisé, l'instance de gestionnaire de files d'attente de secours peut soit retraiter les messages qui ont été supprimés avant l'échec de l'instance active, soit ne pas traiter les messages qui ont été reçus avant l'échec de l'instance active.

Une autre possibilité, rare, est que le système de fichiers contienne l'enregistrement de point de contrôle le plus récent et qu'un journal éloigné non synchronisé sur la base de données de secours ne le contienne pas. Dans ce cas, le gestionnaire de files d'attente ne redémarre pas automatiquement. Vous avez le choix entre attendre que le journal éloigné soit synchronisé ou démarrer à froid le gestionnaire de files d'attente de secours à partir du système de fichiers. Même si, dans ce cas, le système de fichiers contient un point de contrôle plus récent des données du gestionnaire de files d'attente que le journal éloigné, il peut ne pas contenir tous les messages qui ont été traités avant l'échec de l'instance active. Certains messages peuvent être traités à nouveau et d'autres non traités après un redémarrage à froid qui n'est pas synchronisé avec les journaux.

Avec un gestionnaire de files d'attente multi-instance, le système de fichiers est également utilisé pour contrôler l'instance d'un gestionnaire de files d'attente qui est active et celle de secours. L'instance active acquiert un verrou sur les données du gestionnaire de files d'attente. La base de données de secours attend pour acquérir le verrou, et lorsqu'elle le fait, elle devient l'instance active. Le verrou est libéré par l'instance active, s'il se termine normalement. Le verrou est libéré par le système de fichiers si le système de fichiers détecte que l'instance active a échoué ou qu'il ne peut pas accéder au système de fichiers. Le système de fichiers doit répondre aux conditions requises pour la détection des incidents ; voir [Conditions requises pour les systèmes de fichiers partagés](#).

L'architecture des gestionnaires de files d'attente multi-instance sous IBM i permet le redémarrage automatique après une défaillance du serveur ou du gestionnaire de files d'attente. Il prend également en charge la restauration des données du gestionnaire de files d'attente suite à une défaillance du système de fichiers dans lequel les données du gestionnaire de files d'attente sont stockées.

Dans [Figure 36](#), à la [page 307](#), si ALPHA échoue, vous pouvez redémarrer manuellement QM1 en version bêta, à l'aide du journal en miroir. En ajoutant la fonction de gestionnaire de files d'attente multi-instance à QM1, l'instance de secours de QM1 reprend automatiquement sur BETA si l'instance active sur ALPHA échoue. QM1 peut également reprendre automatiquement si c'est le serveur ALPHA qui échoue, et pas seulement l'instance active de QM1. Une fois que BETA devient l'hôte de l'instance de gestionnaire de files d'attente active, l'instance de secours peut être démarrée sur ALPHA.

La [Figure 36](#), à la [page 307](#) présente une configuration qui met en miroir les journaux entre deux instances d'un gestionnaire de files d'attente à l'aide de NetServer pour stocker les données du gestionnaire de files d'attente. Vous pouvez développer le modèle pour inclure plus de journaux, et donc plus d'instances. Suivez les règles de dénomination de journal décrites dans la rubrique «[Journaux de gestionnaire de files d'attente sous IBM i](#)», à la [page 285](#). Actuellement, le nombre d'instances en cours d'exécution d'un gestionnaire de files d'attente est limité à deux, une est active et une est en veille.

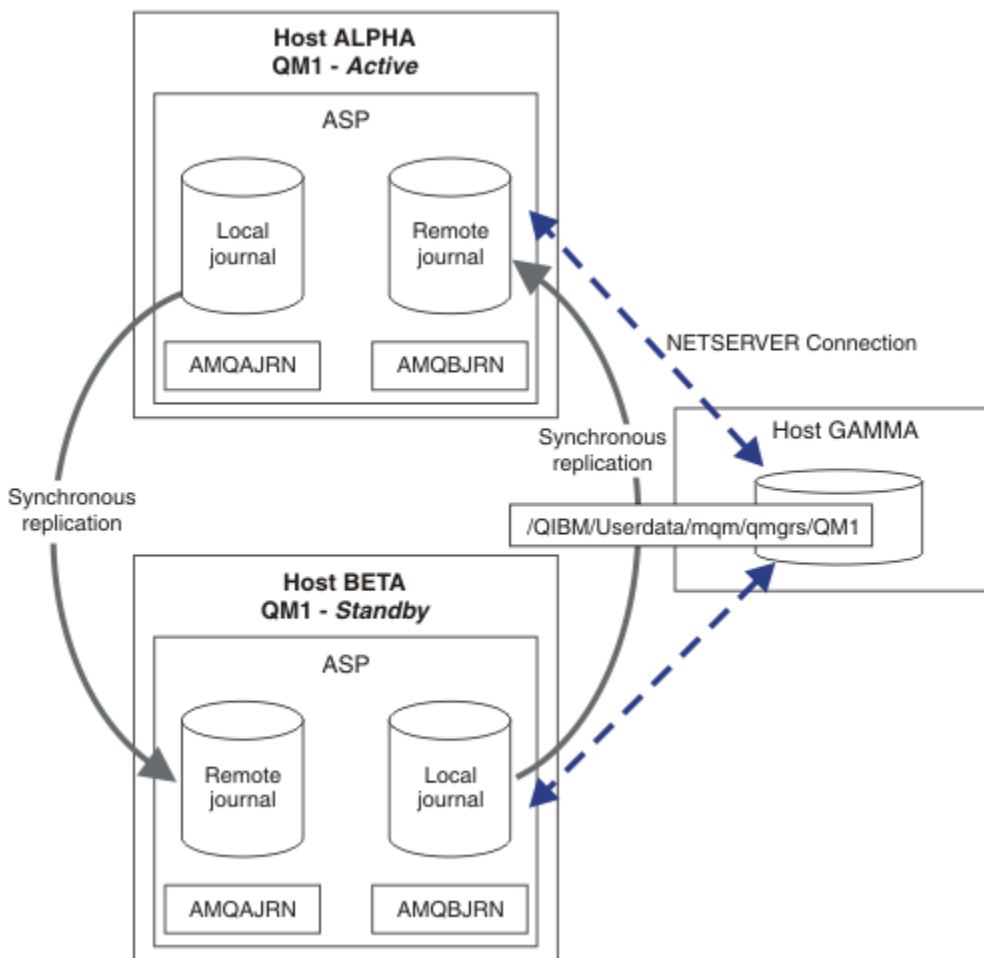


Figure 36. Mettre en miroir un journal de gestionnaire de files d'attente

Le journal local pour QM1 sur l'hôte ALPHA est appelé AMQAJRN (ou plus complètement, QMQM1/AMQAJRN) et sur BETA le journal est QMQM1/AMQBJRN. Chaque journal local est répliqué dans des journaux éloignés sur toutes les autres instances du gestionnaire de files d'attente. Si le gestionnaire de files d'attente est configuré avec deux instances, un journal local est répliqué sur un journal distant.

*SYNC ou *ASYNCRéplication de journal éloigné

Les journaux IBM i sont mis en miroir à l'aide de l'une des deux options suivantes: synchrone (*SYNC) ou asynchrone (*ASYNCR) journalisation; voir [Remote journal management](#).

Le mode de réplication dans [Figure 36](#), à la page 307 est *SYNC, et non *ASYNCR. *ASYNCR est plus rapide, mais si un incident se produit lorsque l'état du journal éloigné est *ASYNCRPEND, le journal local et le journal éloigné ne sont pas cohérents. Le journal éloigné doit rattraper le journal local. Si vous choisissez *SYNC, le système local attend le journal éloigné avant de renvoyer un appel nécessitant une écriture terminée. Les journaux locaux et éloignés restent généralement cohérents les uns avec les autres. Uniquement si l'opération *SYNC prend plus de temps qu'une heure définie¹, et la journalisation à distance est désactivée, les journaux sont désynchronisés. Une erreur est consignée dans la file d'attente de messages du journal et dans QSYSOPR. Le gestionnaire de files d'attente détecte ce message, écrit une erreur dans le journal des erreurs du gestionnaire de files d'attente et désactive la réplication à distance du journal du gestionnaire de files d'attente. L'instance de gestionnaire de files d'attente active reprend sans journalisation à distance dans ce journal. Lorsque le serveur distant est à nouveau

¹ La durée indiquée est de 60 secondes sur IBM i 5 et est comprise entre 1 et 3600 secondes sur IBM i 6.1 et les versions suivantes.

disponible, vous devez réactiver manuellement la réplication de journal distant synchrone. Les journaux sont ensuite resynchronisés.

Un problème lié à la configuration *SYNC / *ASYNC illustrée dans la Figure 36, à la page 307 est la façon dont l'instance de gestionnaire de files d'attente de secours sur BETA prend le contrôle. Dès que l'instance de gestionnaire de files d'attente sur BETA écrit son premier message persistant, elle tente de mettre à jour le journal éloigné sur ALPHA. Si la cause de la transmission du contrôle de ALPHA à BETA est l'échec de ALPHA et que ALPHA est toujours arrêté, la journalisation à distance dans ALPHA échoue. BETA attend la réponse de ALPHA, puis désactive la journalisation à distance et reprend le traitement des messages avec uniquement la journalisation locale. BETA doit attendre un certain temps pour détecter que ALPHA est arrêté, provoquant une période d'inactivité.

Le choix entre la définition de la journalisation à distance sur *SYNC ou *ASYNC est une option de compensation. Le Tableau 22, à la page 308 récapitule les compromis entre l'utilisation de la journalisation *SYNC et *ASYNC entre une paire de gestionnaires de files d'attente:

| Actif | Secours | *SYNC | *ASYNC |
|--------|---------|---|--|
| *SYNC | | <ol style="list-style-type: none"> 1. Commutation et reprise en ligne cohérentes 2. L'instance de secours ne reprend pas immédiatement après la reprise en ligne. 3. La journalisation à distance doit être disponible en permanence 4. Les performances du gestionnaire de files d'attente dépendent de la journalisation à distance | <ol style="list-style-type: none"> 1. Commutation et reprise en ligne cohérentes 2. La journalisation à distance doit être commutée sur *SYNC lorsque le serveur de secours est disponible 3. La journalisation à distance doit rester disponible une fois qu'elle a été redémarrée 4. Les performances du gestionnaire de files d'attente dépendent de la journalisation à distance |
| *ASYNC | | <ol style="list-style-type: none"> 1. Pas une combinaison sensée | <ol style="list-style-type: none"> 1. Certains messages peuvent être perdus ou dupliqués après une reprise en ligne ou une commutation 2. Il n'est pas nécessaire que l'instance de secours soit disponible en permanence pour que l'instance active puisse continuer sans délai. 3. Les performances ne dépendent pas de la journalisation à distance |

*SYNC / *SYNC

L'instance de gestionnaire de files d'attente active utilise la journalisation *SYNC et, lorsque l'instance de gestionnaire de files d'attente de secours démarre, elle tente immédiatement d'utiliser la journalisation *SYNC .

1. Le journal éloigné est cohérent au niveau transactionnel avec le journal local du gestionnaire de files d'attente actif. Si le gestionnaire de files d'attente est basculé sur l'instance de secours, il peut reprendre immédiatement. L'instance de secours reprend normalement sans perte ni duplication des messages. Les messages ne sont perdus ou dupliqués que si la journalisation à distance a échoué depuis le dernier point de contrôle et que le gestionnaire de files d'attente précédemment actif ne peut pas être redémarré.
2. Si le gestionnaire de files d'attente bascule sur l'instance de secours, il se peut qu'il ne puisse pas démarrer immédiatement. L'instance de gestionnaire de files d'attente de secours est activée avec la journalisation *SYNC . La cause de la reprise en ligne peut empêcher la journalisation à distance sur le serveur hébergeant l'instance de secours. Le gestionnaire de files d'attente

attend que le problème soit détecté avant de traiter les messages persistants. Une erreur est consignée dans la file d'attente de messages du journal et dans QSYSOPR. Le gestionnaire de files d'attente détecte ce message, écrit une erreur dans le journal des erreurs du gestionnaire de files d'attente et désactive la réplication à distance du journal du gestionnaire de files d'attente. L'instance de gestionnaire de files d'attente active reprend sans journalisation à distance dans ce journal. Lorsque le serveur distant est à nouveau disponible, vous devez réactiver manuellement la réplication de journal distant synchrone. Les journaux sont ensuite resynchronisés.

3. Le serveur sur lequel le journal distant est répliqué doit toujours être disponible pour gérer le journal distant. Le journal distant est généralement répliqué sur le même serveur que celui qui héberge le gestionnaire de files d'attente de secours. Il se peut que le serveur ne soit plus disponible. Une erreur est consignée dans la file d'attente de messages du journal et dans QSYSOPR. Le gestionnaire de files d'attente détecte ce message, écrit une erreur dans le journal des erreurs du gestionnaire de files d'attente et désactive la réplication à distance du journal du gestionnaire de files d'attente. L'instance de gestionnaire de files d'attente active reprend sans journalisation à distance dans ce journal. Lorsque le serveur distant est à nouveau disponible, vous devez réactiver manuellement la réplication de journal distant synchrone. Les journaux sont ensuite resynchronisés.
4. La journalisation à distance est plus lente que la journalisation locale, et beaucoup plus lente si les serveurs sont séparés par une grande distance. Le gestionnaire de files d'attente doit attendre la journalisation à distance, ce qui réduit les performances du gestionnaire de files d'attente.

La configuration *SYNC / *SYNC entre deux serveurs présente l'inconvénient de retarder la reprise de l'instance de secours après la reprise en ligne. La configuration *SYNC / *ASYNCR ne présente pas cet incident.

*SYNC / *ASYNCR ne garantit aucune perte de message après une commutation ou une reprise en ligne, tant qu'un journal éloigné est disponible. Si vous souhaitez réduire le risque de perte de message après une reprise en ligne ou une commutation, vous disposez de deux options. Arrêtez l'instance active si le journal éloigné devient inactif ou créez des journaux éloignés sur plusieurs serveurs.

***SYNC / *ASYNCR**

L'instance de gestionnaire de files d'attente active utilise la journalisation *SYNC et, lorsque l'instance de gestionnaire de files d'attente de secours démarre, elle utilise la journalisation *ASYNCR. Peu de temps après la disponibilité du serveur hébergeant la nouvelle instance de secours, l'opérateur système doit basculer le journal éloigné sur l'instance active vers *SYNC. Lorsque l'opérateur fait passer la journalisation éloignée de *ASYNCR à *SYNC, l'instance active s'interrompt si l'état du journal éloigné est *ASYNCRPEND. L'instance de gestionnaire de files d'attente active attend que les postes de journal restants soient transférés vers le journal éloigné. Lorsque le journal éloigné a été synchronisé avec le journal local, la nouvelle base de données de secours est à nouveau cohérente au niveau transactionnel avec la nouvelle instance active. Du point de vue de la gestion des gestionnaires de files d'attente multi-instance, dans une configuration *SYNC / *ASYNCR, l'opérateur système IBM i a une tâche supplémentaire. L'opérateur doit basculer la journalisation à distance vers *SYNC en plus du redémarrage de l'instance de gestionnaire de files d'attente défaillante.

1. Le journal éloigné est cohérent au niveau transactionnel avec le journal local du gestionnaire de files d'attente actif. Si l'instance de gestionnaire de files d'attente active est basculée ou bascule sur l'instance de secours, l'instance de secours peut reprendre immédiatement. L'instance de secours reprend normalement sans perte ni duplication des messages. Les messages ne sont perdus ou dupliqués que si la journalisation à distance a échoué depuis le dernier point de contrôle et que le gestionnaire de files d'attente précédemment actif ne peut pas être redémarré.
2. L'opérateur système doit basculer le journal éloigné de *ASYNCR vers *SYNC peu après que le système hébergeant l'instance active soit à nouveau disponible. L'opérateur peut attendre le rattrapage du journal éloigné avant de passer à *SYNC. L'opérateur peut également basculer immédiatement l'instance distante sur *SYNC et forcer l'instance active à attendre que le journal de l'instance de secours soit rattrapé. Lorsque la journalisation à distance est définie sur *SYNC, l'instance de secours est généralement cohérente au niveau transactionnel avec l'instance active. Les messages ne sont perdus ou dupliqués que si la journalisation à distance a échoué depuis le dernier point de contrôle et que le gestionnaire de files d'attente précédemment actif ne peut pas être redémarré.

3. Lorsque la configuration a été restaurée à partir d'une commutation ou d'une reprise en ligne, le serveur sur lequel le journal éloigné est hébergé doit être disponible en permanence.

Choisissez *SYNC / *ASYNC lorsque vous souhaitez que le gestionnaire de files d'attente de secours reprenne rapidement après une reprise en ligne. Vous devez restaurer manuellement le paramètre de journal éloigné sur *SYNC sur la nouvelle instance active. La configuration *SYNC / *ASYNC correspond au modèle normal d'administration d'une paire de gestionnaires de files d'attente multi-instance. Une fois qu'une instance a échoué, il y a un délai avant le redémarrage de l'instance de secours, au cours duquel l'instance active ne peut pas basculer.

***ASYNC / *ASYNC**

Les deux serveurs hébergeant les gestionnaires de files d'attente actifs et de secours sont configurés pour utiliser la journalisation à distance *ASYNC .

1. Lors de la commutation ou de la reprise en ligne, le gestionnaire de files d'attente continue avec le journal sur le nouveau serveur. Il se peut que le journal ne soit pas synchronisé lors de la commutation ou de la reprise en ligne. Par conséquent, des messages peuvent être perdus ou dupliqués.
2. L'instance active s'exécute, même si le serveur hébergeant le gestionnaire de files d'attente de secours n'est pas disponible. Le journal local est répliqué de manière asynchrone avec le serveur de secours lorsqu'il est disponible.
3. Les performances du gestionnaire de files d'attente local ne sont pas affectées par la journalisation à distance.

Choisissez *ASYNC / *ASYNC si les performances sont votre exigence principale et que vous êtes prêt à perdre ou à dupliquer certains messages après la reprise en ligne ou la commutation.

***ASYNC / *SYNC**

Il n'y a aucune raison d'utiliser cette combinaison d'options.

Activation du gestionnaire de files d'attente à partir d'un journal éloigné

Les journaux sont répliqués de manière synchrone ou asynchrone. Il se peut que le journal éloigné ne soit pas actif ou qu'il rattrape le journal local. Le journal distant peut être en cours de rattrapage, même s'il est répliqué de manière synchrone, car il a peut-être été récemment activé. Les règles que le gestionnaire de files d'attente applique à l'état du journal éloigné qu'il utilise lors du démarrage sont les suivantes.

1. Le démarrage de la base de données de secours échoue si elle doit être relecture à partir du journal éloigné sur la base de données de secours et que le statut du journal est *FAILED ou *INACTPEND.
2. Lorsque l'activation de la base de données de secours commence, l'état du journal éloigné sur la base de données de secours doit être *ACTIVE ou *INACTIVE. Si l'état est *INACTIVE, l'activation peut échouer si toutes les données de journal n'ont pas été répliquées.

L'échec se produit si les données du gestionnaire de files d'attente sur le système de fichiers réseau contiennent un enregistrement de point de contrôle plus récent que celui présent dans le journal éloigné. Il est peu probable que l'échec se produise tant que le journal éloigné est bien activé dans l'intervalle maximal de 30 minutes par défaut entre les points de contrôle. Si le gestionnaire de files d'attente de secours lit un enregistrement de point de contrôle plus récent à partir du système de fichiers, il ne démarre pas.

Vous avez le choix: attendez que le journal local sur le serveur actif puisse être restauré ou démarrez à froid le gestionnaire de files d'attente de secours. Si vous choisissez le démarrage à froid, le gestionnaire de files d'attente démarre sans données de journal et repose sur la cohérence et l'exhaustivité des données du gestionnaire de files d'attente dans le système de fichiers.

Remarque : Si vous démarrez un gestionnaire de files d'attente à froid, vous risquez de perdre ou de dupliquer des messages après le dernier point de contrôle. Les transactions de message ont été écrites dans le journal, mais certaines des transactions n'ont peut-être pas été écrites dans les données du gestionnaire de files d'attente dans le système de fichiers. Lorsque vous démarrez un gestionnaire de files d'attente à froid, un nouveau journal est démarré et les transactions qui ne sont pas écrites dans les données du gestionnaire de files d'attente du système de fichiers sont perdues.

3. L'activation du gestionnaire de files d'attente de secours attend que l'état du journal éloigné sur le serveur de secours passe de *ASYNCPEND ou *SYNCPEND à *ASYNC ou *SYNC. Des messages sont écrits périodiquement dans le journal des travaux du contrôleur d'exécution.

Remarque : Dans ce cas, l'activation est en attente sur le journal éloigné local du gestionnaire de files d'attente de secours en cours d'activation. Le gestionnaire de files d'attente attend également un certain temps avant de continuer sans journal éloigné. Il attend lorsqu'il tente d'écrire de manière synchrone dans son ou ses journaux éloignés et que le journal n'est pas disponible.

4. L'activation s'arrête si le journal passe à l'état *FAILED ou *INACTPEND.

Les noms et les états des journaux locaux et éloignés à utiliser dans l'activation sont consignés dans le journal des erreurs du gestionnaire de files d'attente.

IBM i *Création d'un gestionnaire de files d'attente multi-instance à l'aide de la mise en miroir du journal et de NetServer sous IBM i*

Créez un gestionnaire de files d'attente multi-instance à exécuter sur deux serveurs IBM i . Les données du gestionnaire de files d'attente sont stockées sur un troisième serveur IBM i à l'aide de NetServer. Le journal du gestionnaire de files d'attente est mis en miroir entre les deux serveurs à l'aide de la journalisation éloignée. La commande **ADDQMJRN** permet de simplifier la création des journaux éloignés.

Avant de commencer

1. La tâche requiert trois serveurs IBM i . Installez IBM MQ sur deux d'entre eux, ALPHA et BETA dans l'exemple. IBM MQ doit être au moins à la version 7.0.1.1.
2. Le troisième serveur est un serveur IBM i , connecté par NetServer à ALPHA et BETA. Il est utilisé pour partager les données du gestionnaire de files d'attente. Il n'est pas nécessaire d'avoir une installation IBM MQ . Il est utile d'installer IBM MQ sur le serveur en tant qu'étape temporaire pour configurer les répertoires et les droits du gestionnaire de files d'attente.
3. Assurez-vous que le profil utilisateur QMQM possède le même mot de passe sur les trois serveurs.
4. Installez IBM i NetServer; voir [i5/OS NetServer](#).

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour créer la configuration illustrée dans la [Figure 37](#), à la [page 314](#). Les données du gestionnaire de files d'attente sont connectées à l'aide d' IBM i NetServer.

- Créez des connexions depuis ALPHA et BETA vers le partage de répertoire sur GAMMA qui permet de stocker les données du gestionnaire de files d'attente. La tâche configure également les droits d'accès, les profils utilisateur et les mots de passe nécessaires.
- Ajoutez des entrées de base de données relationnelle (RDBE) aux systèmes IBM i qui vont exécuter des instances de gestionnaire de files d'attente. Les entrées RDBE sont utilisées pour la connexion aux systèmes IBM i utilisés pour la journalisation à distance.
- Créez le gestionnaire de files d'attente QM1 sur le serveur IBM i , ALPHA.
- Ajoutez les informations de contrôle du gestionnaire de files d'attente pour QM1 sur l'autre serveur IBM i , BETA.
- Créez des journaux éloignés sur les deux serveurs IBM i pour les deux instances de gestionnaire de files d'attente. Chaque gestionnaire de files d'attente écrit dans le journal local. Le journal local est répliqué dans le journal éloigné. La commande **ADDQMJRN** simplifie l'ajout des journaux et des connexions.
- Démarrez le gestionnaire de files d'attente en autorisant une instance de secours.

Procédure

1. Exécutez la tâche «[Création d'un partage de réseau pour les données de gestionnaire de files d'attente à l'aide de NetServer sous IBM i](#)», à la [page 297](#).

Par conséquent, ALPHA et BETA ont un partage, /QNTC/GAMMA/WMQ, qui pointe vers /QIBM/UserData/mqm/qmgrs sur GAMMA. Les profils utilisateur QMQM et QMQMADM disposent des droits nécessaires et QMQM possède des mots de passe correspondants sur les trois systèmes.

2. Ajoutez des entrées de base de données relationnelle (RDBE) aux systèmes IBM i qui vont héberger des instances de gestionnaire de files d'attente.

a) Sur ALPHA, créez la connexion à BETA.

```
ADDRDBDIRE RDB(BETA) RMTLOCNAME(BETA *IP) RMTAUTMTH(*USRIDPWD)
```

b) Sur BETA, créez les connexions à ALPHA.

```
ADDRDBDIRE RDB(ALPHA) RMTLOCNAME(ALPHA *IP) RMTAUTMTH(*USRIDPWD)
```

3. Créez le gestionnaire de files d'attente QM1 sur ALPHA, en sauvegardant les données du gestionnaire de files d'attente sur GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIRP(' /QNTC/GAMMA/WMQ')
```

Le chemin d'accès utilise NetServer pour créer les données du gestionnaire de files d'attente.

4. Exécuter sur ALPHA. La commande ajoute un journal éloigné sur BETA pour.

```
ADDMQMJRN MQMNAME(QM1) RMTJRN RDB(BETA)
```

crée des entrées de journal dans son journal local sur ALPHA lorsque l'instance active est sur ALPHA. Le journal local sur ALPHA est répliqué sur le journal éloigné sur BETA.

5. Utilisez la commande, pour inspecter les données de configuration IBM MQ créées par pour sur ALPHA.

Les informations sont nécessaires à l'étape suivante.

Dans cet exemple, la configuration suivante est créée dans ALPHA pour:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

6. Créez une instance de gestionnaire de files d'attente de QM1 sur BETA à l'aide de la commande. Exécutez la commande suivante sur BETA pour modifier les informations de contrôle du gestionnaire de files d'attente sur BETA.

```
ADDMQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMOM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1')
```

Conseil : Copiez et collez les informations de configuration. La section du gestionnaire de files d'attente est la même sur ALPHA et BETA.

7. Exécuter sur BETA. La commande ajoute un journal local sur BETA et un journal éloigné sur ALPHA pour.

```
ADDMQMJRN MQMNAME(QM1) RMTJRN RDB(ALPHA)
```


crée des entrées de journal dans son journal local sur BETA lorsque l'instance active est sur BETA. Le journal local sur BETA est répliqué sur le journal éloigné sur ALPHA.

Remarque : Vous pouvez également configurer la journalisation à distance de BETA à ALPHA à l'aide de la journalisation asynchrone.

Utilisez cette commande pour configurer la journalisation asynchrone de BETA vers ALPHA, à la place de la commande de l'étape «7», à la page 312.

```
ADDMQMJRN MQMNAME (QM1) RMTJRNRDB (ALPHA) RMTJRNDLV (*ASYNC)
```

Si le serveur ou la journalisation sur ALPHA est la source de l'échec, la fonction BETA démarre sans attendre que de nouvelles entrées de journal soient répliquées sur ALPHA.

Basculez le mode de réplication sur *SYNC, à l'aide de la commande, lorsque ALPHA est à nouveau en ligne.

Utilisez les informations de la rubrique «[Configuration de journal protégé par disque miroir pour ASP sous IBM i](#)», à la page 305 pour déterminer si les journaux doivent être mis en miroir de manière synchrone, asynchrone ou mixte. Par défaut, la réplication est synchrone, avec une période d'attente de 60 secondes pour une réponse du journal éloigné.

8. Vérifiez que les journaux ALPHA et BETA sont activés et que le statut de la réplication de journal éloigné est activé.

a) Sur ALPHA:

```
WRKMQMJRN MQMNAME (QM1)
```

b) Sur BETA:

```
WRKMQMJRN MQMNAME (QM1)
```

9. Démarrez les instances de gestionnaire de files d'attente sur ALPHA et BETA.

a) Démarrez la première instance sur ALPHA, ce qui en fait l'instance active. Activation du basculement vers une instance de secours.

```
STRMQM MQMNAME (QM1) STANDBY (*YES)
```

b) Démarrez la deuxième instance sur BETA, ce qui en fait l'instance de secours.

```
STRMQM MQMNAME (QM1) STANDBY (*YES)
```

Résultats

Permet de vérifier le statut du gestionnaire de files d'attente:

1. Le statut de l'instance de gestionnaire de files d'attente sur ALPHA doit être.
2. Le statut de l'instance de gestionnaire de files d'attente sur BETA doit être.

Exemple

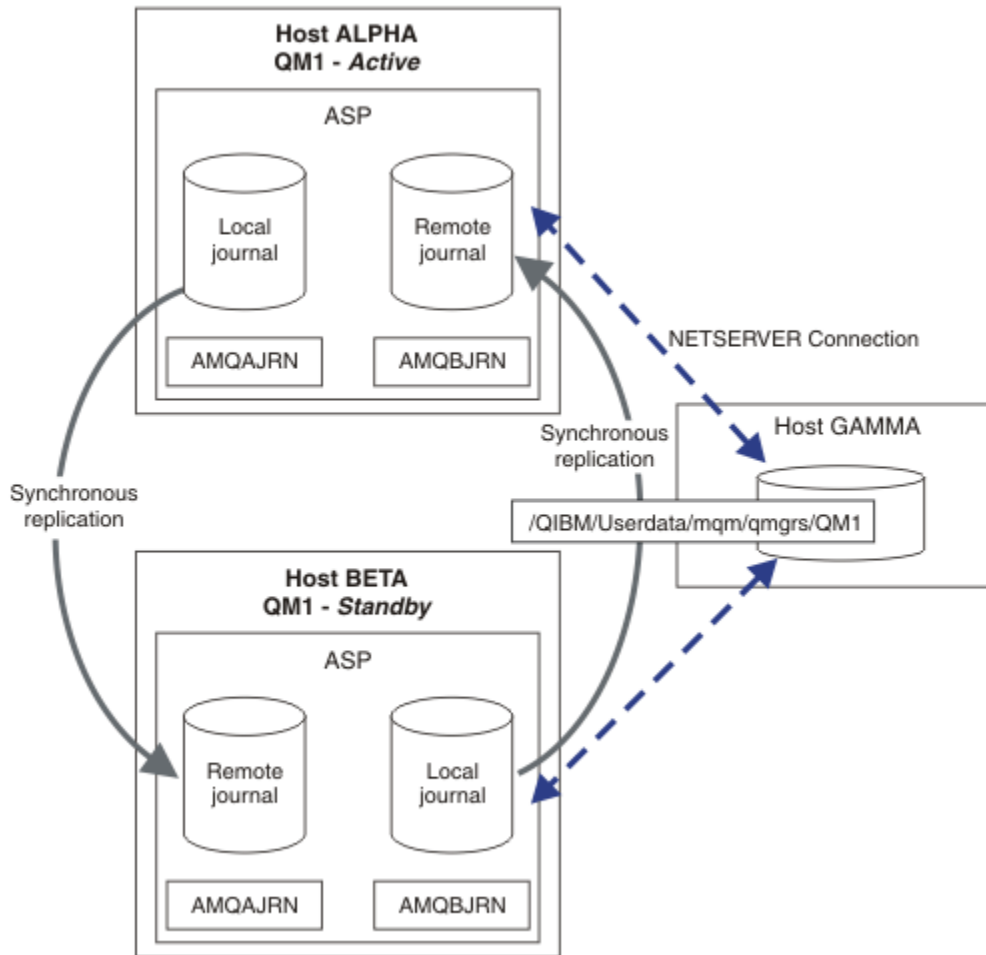


Figure 37. Configuration du journal en miroir

Que faire ensuite

- Vérifiez que les instances actives et de secours basculent automatiquement. Vous pouvez exécuter les exemples de programmes à haute disponibilité pour tester le basculement ; voir [Exemples de programmes à haute disponibilité](#). Les exemples de programme sont des clients 'C'. Vous pouvez les exécuter à partir d'une plateforme Windows ou Unix.

1. Démarrez les exemples de programmes à haute disponibilité.
2. Sous ALPHA, arrêtez le gestionnaire de files d'attente demandant le basculement:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Vérifiez que l'instance de sur BETA est active.
4. Redémarrer sur ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Examinez les autres configurations à haute disponibilité:
 1. Utilisez NetServer pour placer les données du gestionnaire de files d'attente sur un serveur Windows .

2. Au lieu d'utiliser la journalisation éloignée pour reproduire le journal du gestionnaire de files d'attente, stockez le journal sur un ASP indépendant. Utilisez la mise en grappe IBM i pour transférer l'ASP indépendant de ALPHA vers BETA.

IBM i Conversion d'un gestionnaire de files d'attente à instance unique en gestionnaire de files d'attente multi-instance à l'aide de NetServer et de la mise en miroir du journal sous IBM i

Convertissez un gestionnaire de files d'attente à instance unique en gestionnaire de files d'attente multi-instance. Déplacez les données du gestionnaire de files d'attente vers un partage de réseau connecté par NetServer. Reproduisez le journal du gestionnaire de files d'attente sur un second serveur IBM i à l'aide de la journalisation à distance.

Avant de commencer

1. La tâche requiert trois serveurs IBM i . L'installation IBM MQ existante, sur le serveur ALPHA dans l'exemple, doit être au moins au niveau de IBM WebSphere MQ 7.0.1 Fix Pack 1. ALPHA exécute un gestionnaire de files d'attente appelé QM1 dans l'exemple.
2. Installez IBM MQ sur le second serveur IBM i , BETA dans l'exemple.
3. Le troisième serveur est un serveur IBM i , connecté par NetServer à ALPHA et BETA. Il est utilisé pour partager les données du gestionnaire de files d'attente. Il n'est pas nécessaire d'avoir une installation IBM MQ . Il est utile d'installer IBM MQ sur le serveur en tant qu'étape temporaire pour configurer les répertoires et les droits du gestionnaire de files d'attente.
4. Assurez-vous que le profil utilisateur QMQM possède le même mot de passe sur les trois serveurs.
5. Installez IBM i NetServer; voir [i5/OS NetServer](#).

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour convertir un gestionnaire de files d'attente à instance unique en gestionnaire de files d'attente multi-instance indiqué dans la [Figure 38](#), à la [page 319](#). Le gestionnaire de files d'attente à instance unique est supprimé dans la tâche, puis recréé, stockant les données du gestionnaire de files d'attente sur le partage de réseau connecté par NetServer. Cette procédure est plus fiable que le déplacement des répertoires et des fichiers du gestionnaire de files d'attente vers le partage de réseau à l'aide de la commande **CPY** .

- Créez des connexions depuis ALPHA et BETA vers le partage de répertoire sur GAMMA qui permet de stocker les données du gestionnaire de files d'attente. La tâche configure également les droits d'accès, les profils utilisateur et les mots de passe nécessaires.
- Ajoutez des entrées de base de données relationnelle (RDBE) aux systèmes IBM i qui vont exécuter des instances de gestionnaire de files d'attente. Les entrées RDBE sont utilisées pour la connexion aux systèmes IBM i utilisés pour la journalisation à distance.
- Sauvegardez les journaux et les définitions du gestionnaire de files d'attente, arrêtez le gestionnaire de files d'attente et supprimez-le.
- Recréez le gestionnaire de files d'attente en stockant les données du gestionnaire de files d'attente sur le partage de réseau sur GAMMA.
- Ajoutez la deuxième instance du gestionnaire de files d'attente à l'autre serveur.
- Créez des journaux éloignés sur les deux serveurs IBM i pour les deux instances de gestionnaire de files d'attente. Chaque gestionnaire de files d'attente écrit dans le journal local. Le journal local est répliqué dans le journal éloigné. La commande **ADDMQMJRN** simplifie l'ajout des journaux et des connexions.
- Démarrez le gestionnaire de files d'attente en autorisant une instance de secours.

Remarque :

A l'étape «4», à la [page 316](#) de la tâche, vous supprimez le gestionnaire de files d'attente à instance unique, QM1. La suppression du gestionnaire de files d'attente supprime tous les messages persistants dans les files d'attente. Pour cette raison, terminez le traitement de tous les messages stockés par le gestionnaire de files d'attente, avant de convertir le gestionnaire de files d'attente. Si le traitement de tous les messages n'est pas possible, sauvegardez la bibliothèque du gestionnaire de files d'attente avant

l'étape «4», à la page 316. Restaurez la bibliothèque du gestionnaire de files d'attente après l'étape «5», à la page 316.

Remarque :

A l'étape «5», à la page 316 de la tâche, vous recréez QM1. Bien que le gestionnaire de files d'attente ait le même nom, il possède un identificateur de gestionnaire de files d'attente différent. La mise en cluster du gestionnaire de files d'attente utilise l'identificateur de gestionnaire de files d'attente. Pour supprimer et recréer un gestionnaire de files d'attente dans un cluster, vous devez d'abord supprimer le gestionnaire de files d'attente du cluster ; voir [Suppression d'un gestionnaire de files d'attente d'un cluster: méthode alternative](#) ou [Suppression d'un gestionnaire de files d'attente d'un cluster](#). Une fois que vous avez recréé le gestionnaire de files d'attente, ajoutez-le au cluster. Bien qu'il porte le même nom qu'auparavant, il semble s'agir d'un nouveau gestionnaire de files d'attente pour les autres gestionnaires de files d'attente du cluster.

Procédure

1. Exécutez la tâche «Création d'un partage de réseau pour les données de gestionnaire de files d'attente à l'aide de NetServer sous IBM i», à la page 297.

Par conséquent, ALPHA et BETA ont un partage, /QNTC/GAMMA/WMQ, qui pointe vers /QIBM/UserData/mqm/qmgrs sur GAMMA. Les profils utilisateur QMQM et QMQMADM disposent des droits nécessaires et QMQM possède des mots de passe correspondants sur les trois systèmes.

2. Ajoutez des entrées de base de données relationnelle (RDBE) aux systèmes IBM i qui vont héberger des instances de gestionnaire de files d'attente.
 - a) Sur ALPHA, créez la connexion à BETA.

```
ADDRDBDIRE RDB(BETA) RMTLOCNAME(BETA *IP) RMTAUTMTH(*USRIDPWD)
```

- b) Sur BETA, créez les connexions à ALPHA.

```
ADDRDBDIRE RDB(ALPHA) RMTLOCNAME(ALPHA *IP) RMTAUTMTH(*USRIDPWD)
```

3. Créez les scripts qui recréent les objets de gestionnaire de files d'attente.

```
QSAVEQMGR LCLQMGRNAM(QM1) FILENAME('*CURLIB/QMQSC(QM1)')  
OUTPUT(*REPLACE) MAKEAUTH(*YES) AUTHFN('*CURLIB/QMAUT(QM1)')
```

4. Arrêtez le gestionnaire de files d'attente et supprimez-le.

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ENDCCTJOB(*YES) RCDMQMIMG(*YES) TIMEOUT(15)  
DLTMQM MQMNAME(QM1)
```

5. Créez le gestionnaire de files d'attente QM1 sur ALPHA, en sauvegardant les données du gestionnaire de files d'attente sur GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)  
MQMDIRP('/QNTC/GAMMA/WMQ')
```

Le chemin d'accès utilise NetServer pour créer les données du gestionnaire de files d'attente.

6. Recréez les objets de gestionnaire de files d'attente pour QM1 à partir des définitions sauvegardées.

```
STRMQMQSC SRCMBR(QM1) SRCFILE(*CURLIB/QMQSC) MQMNAME(QM1)
```

7. Appliquez les autorisations à partir des informations sauvegardées.

- a) Compilez le programme d'autorisation sauvegardé.

```
CRTCLPGM PGM(*CURLIB/QM1) SRCFILE(*CURLIB/QMAUT)
SRCMBR(QM1) REPLACE(*YES)
```

- b) Exécutez le programme pour appliquer les autorisations.

```
CALL PGM(*CURLIB/QM1)
```

- c) Actualisez les informations de sécurité pour QM1.

```
RFRMQMAUT MQMNAME(QM1)
```

8. Exécuter sur ALPHA. La commande ajoute un journal éloigné sur BETA pour.

```
ADDQMJRN MQMNAME(QM1) RMTJRNRDB(BETA)
```

créé des entrées de journal dans son journal local sur ALPHA lorsque l'instance active est sur ALPHA. Le journal local sur ALPHA est répliqué sur le journal éloigné sur BETA.

9. Utilisez la commande, pour inspecter les données de configuration IBM MQ créées par pour sur ALPHA.

Les informations sont nécessaires à l'étape suivante.

Dans cet exemple, la configuration suivante est créée dans ALPHA pour:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

10. Créez une instance de gestionnaire de files d'attente de QM1 sur BETA à l'aide de la commande. Exécutez la commande suivante sur BETA pour modifier les informations de contrôle du gestionnaire de files d'attente sur BETA.

```
ADDQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1')
```

Conseil : Copiez et collez les informations de configuration. La section du gestionnaire de files d'attente est la même sur ALPHA et BETA.

11. Exécuter sur BETA. La commande ajoute un journal local sur BETA et un journal éloigné sur ALPHA pour.

```
ADDQMJRN MQMNAME(QM1) RMTJRNRDB(ALPHA)
```

créé des entrées de journal dans son journal local sur BETA lorsque l'instance active est sur BETA. Le journal local sur BETA est répliqué sur le journal éloigné sur ALPHA.

Remarque : Vous pouvez également configurer la journalisation à distance de BETA à ALPHA à l'aide de la journalisation asynchrone.

Utilisez cette commande pour configurer la journalisation asynchrone de BETA vers ALPHA, à la place de la commande de l'étape «7», à la page 312.

```
ADDQMJRNLV MQMNAME(QM1) RMTJRNRDB(ALPHA) RMTJRNDLV(*ASYNCR)
```

Si le serveur ou la journalisation sur ALPHA est la source de l'échec, la fonction BETA démarre sans attendre que de nouvelles entrées de journal soient répliquées sur ALPHA.

Basculez le mode de répllication sur *SYNC, à l'aide de la commande, lorsque ALPHA est à nouveau en ligne.

Utilisez les informations de la rubrique «[Configuration de journal protégé par disque miroir pour ASP sous IBM i](#)», à la page 305 pour déterminer si les journaux doivent être mis en miroir de manière synchrone, asynchrone ou mixte. Par défaut, la répllication est synchrone, avec une période d'attente de 60 secondes pour une réponse du journal éloigné.

12. Vérifiez que les journaux ALPHA et BETA sont activés et que le statut de la répllication de journal éloigné est activé.

a) Sur ALPHA:

```
WRKMQMJRN MQMNAME(QM1)
```

b) Sur BETA:

```
WRKMQMJRN MQMNAME(QM1)
```

13. Démarrez les instances de gestionnaire de files d'attente sur ALPHA et BETA.

a) Démarrez la première instance sur ALPHA, ce qui en fait l'instance active. Activation du basculement vers une instance de secours.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

b) Démarrez la deuxième instance sur BETA, ce qui en fait l'instance de secours.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

Résultats

Permet de vérifier le statut du gestionnaire de files d'attente:

1. Le statut de l'instance de gestionnaire de files d'attente sur ALPHA doit être.
2. Le statut de l'instance de gestionnaire de files d'attente sur BETA doit être.

Exemple

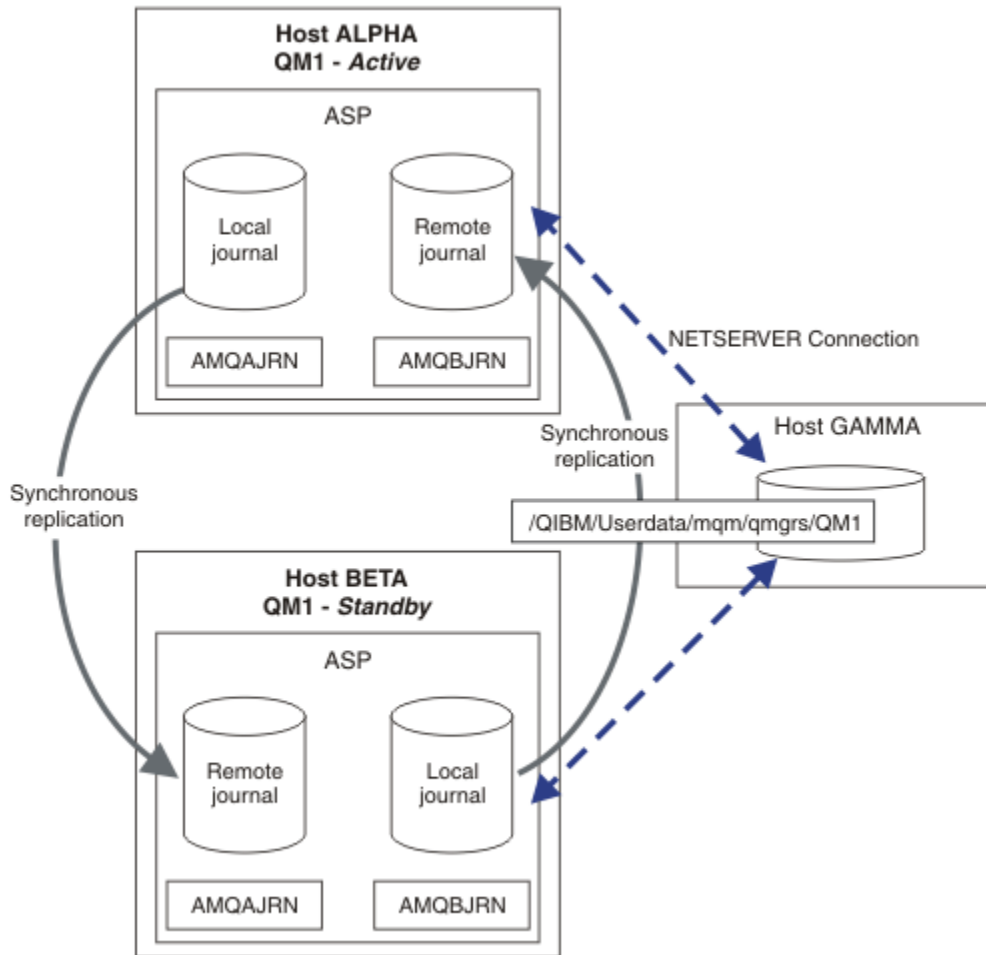


Figure 38. Configuration du journal en miroir

Que faire ensuite

- Vérifiez que les instances actives et de secours basculent automatiquement. Vous pouvez exécuter les exemples de programmes à haute disponibilité pour tester le basculement ; voir [Exemples de programmes à haute disponibilité](#). Les exemples de programme sont des clients 'C'. Vous pouvez les exécuter à partir d'une plateforme Windows ou Unix.

1. Démarrez les exemples de programmes à haute disponibilité.
2. Sous ALPHA, arrêtez le gestionnaire de files d'attente demandant le basculement:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Vérifiez que l'instance de sur BETA est active.
4. Redémarrer sur ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Examinez les autres configurations à haute disponibilité:
 1. Utilisez NetServer pour placer les données du gestionnaire de files d'attente sur un serveur Windows .

2. Au lieu d'utiliser la journalisation éloignée pour reproduire le journal du gestionnaire de files d'attente, stockez le journal sur un ASP indépendant. Utilisez la mise en grappe IBM i pour transférer l'ASP indépendant de ALPHA vers BETA.

IBM i Configuration de journal ASP indépendant commuté sous IBM i

Il n'est pas nécessaire de répliquer un journal ASP indépendant pour créer une configuration de gestionnaire de files d'attente multi-instance. Vous devez automatiser un moyen de transférer l'ASP indépendant du gestionnaire de files d'attente actif vers le gestionnaire de files d'attente de secours. Il existe d'autres solutions de haute disponibilité possibles à l'aide d'un ASP indépendant, qui ne nécessitent pas toutes l'utilisation d'un gestionnaire de files d'attente multi-instance.

Lorsque vous utilisez un ASP indépendant, vous n'avez pas besoin de mettre en miroir le journal du gestionnaire de files d'attente. Si vous avez installé la gestion de cluster et que les serveurs hébergeant les instances de gestionnaire de files d'attente se trouvent dans le même groupe de ressources en cluster, le journal du gestionnaire de files d'attente peut être transféré automatiquement à un autre serveur à une courte distance du serveur actif, si l'hôte exécutant l'instance active échoue. Vous pouvez également transférer le journal manuellement, dans le cadre d'un commutateur planifié, ou vous pouvez écrire une procédure de commande pour transférer l'ASP indépendant à l'aide d'un programme.

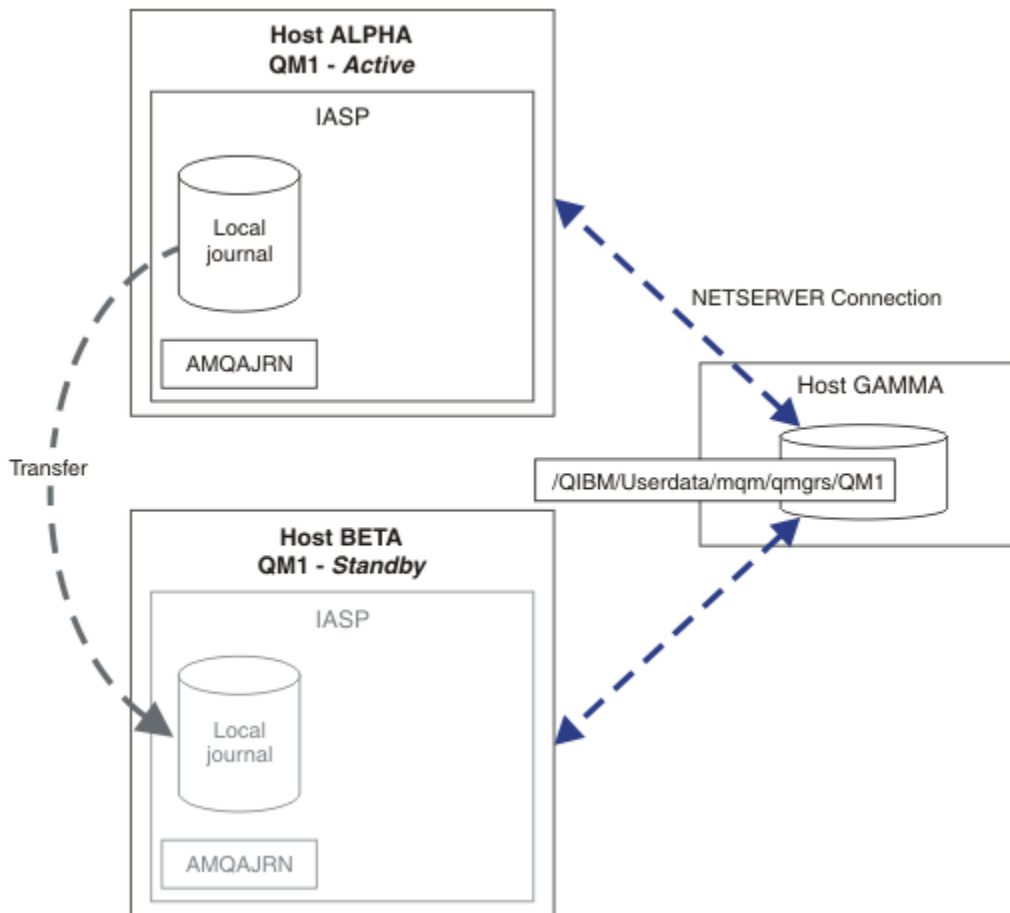


Figure 39. Transfert d'un journal de gestionnaire de files d'attente à l'aide d'un ASP indépendant

Pour le fonctionnement d'un gestionnaire de files d'attente multi-instance, les données du gestionnaire de files d'attente doivent être stockées sur un système de fichiers partagé. Le système de fichiers peut être hébergé sur différentes plateformes. Vous ne pouvez pas stocker de données de gestionnaire de files d'attente multi-instance sur un ASP ou un ASP indépendant.

Le système de fichiers partagé joue deux rôles dans la configuration: les mêmes données de gestionnaire de files d'attente sont partagées entre toutes les instances du gestionnaire de files d'attente. Le système

de fichiers doit disposer d'un protocole de verrouillage robuste qui garantit qu'une seule instance du gestionnaire de files d'attente a accès aux données du gestionnaire de files d'attente une fois qu'elle a démarré. Si le gestionnaire de files d'attente est défaillant ou que les communications avec le serveur de fichiers sont rompées, le système de fichiers doit libérer le verrou sur les données du gestionnaire de files d'attente détenues par l'instance active qui ne communique plus avec le système de fichiers. L'instance de gestionnaire de files d'attente de secours peut alors accéder en lecture / écriture aux données du gestionnaire de files d'attente. Le protocole de système de fichiers doit être conforme à un ensemble de règles pour fonctionner correctement avec les gestionnaires de files d'attente multi-instance (voir «Composants d'une solution à haute disponibilité sur IBM i», à la page 296).

Le mécanisme de verrouillage sérialise la commande de démarrage du gestionnaire de files d'attente et contrôle l'instance du gestionnaire de files d'attente qui est active. Une fois qu'un gestionnaire de files d'attente devient actif, il reconstruit ses files d'attente à partir du journal local que vous ou le cluster à haute disponibilité avez transféré vers le serveur de secours. Les clients reconnectables en attente de reconnexion au même gestionnaire de files d'attente sont reconnectés et toutes les transactions en cours sont annulées. Les applications configurées pour démarrer en tant que services de gestionnaire de files d'attente sont démarrées.

Vous devez vous assurer que le journal local de l'instance de gestionnaire de files d'attente active qui a échoué sur l'ASP indépendant est transféré sur le serveur qui héberge l'instance de gestionnaire de files d'attente de secours qui vient d'être activée, soit en configurant le gestionnaire de ressources en grappe, soit en transférant manuellement l'ASP indépendant. L'utilisation d'ASP indépendants n'empêche pas la configuration de journaux éloignés et de la mise en miroir, si vous décidez d'utiliser un ASP indépendant pour la sauvegarde et la reprise après incident, et d'utiliser la mise en miroir de journaux éloignés pour la configuration du gestionnaire de files d'attente multi-instance.

Si vous avez choisi d'utiliser un ASP indépendant, vous pouvez envisager d'autres configurations à haute disponibilité. L'arrière-plan de ces solutions est décrit dans «ASP indépendants et haute disponibilité», à la page 324.

1. Au lieu d'utiliser des gestionnaires de files d'attente multi-instance, installez et configurez un gestionnaire de files d'attente à instance unique entièrement sur un ASP indépendant, et utilisez les services de haute disponibilité IBM i pour faire basculer le gestionnaire de files d'attente. Vous devrez probablement augmenter la solution avec un moniteur de gestionnaire de files d'attente pour détecter si le gestionnaire de files d'attente a échoué indépendamment du serveur. Il s'agit de la base de la solution fournie dans *Supportpac MC41: Configuration de IBM MQ for iSeries for High Availability*.
2. Utilisez des ASP indépendants et la protection par disque miroir multisite (XSM) pour mettre en miroir l'ASP indépendant au lieu de le permuter sur le bus local. Cela étend la plage géographique de la solution ASP indépendante jusqu'à ce que le temps nécessaire à l'écriture des enregistrements de journal sur une longue distance le permette.

Création d'un gestionnaire de files d'attente multi-instance à l'aide d'un ASP indépendant et de NetServer sous IBM i

Créez un gestionnaire de files d'attente multi-instance à exécuter sur deux serveurs IBM i . Les données du gestionnaire de files d'attente sont stockées sur un serveur IBM i à l'aide de NetServer. Le journal du gestionnaire de files d'attente est stocké dans un ASP indépendant. Utilisez la mise en grappe IBM i ou une procédure manuelle pour transférer l'ASP indépendant contenant le journal du gestionnaire de files d'attente vers l'autre serveur IBM i .

Avant de commencer

1. La tâche requiert trois serveurs IBM i . Installez IBM MQ sur deux d'entre eux, ALPHA et BETA dans l'exemple. IBM MQ doit être au moins à la version 7.0.1.1.
2. Le troisième serveur est un serveur IBM i , connecté par NetServer à ALPHA et BETA. Il est utilisé pour partager les données du gestionnaire de files d'attente. Il n'est pas nécessaire d'avoir une installation IBM MQ . Il est utile d'installer IBM MQ sur le serveur en tant qu'étape temporaire pour configurer les répertoires et les droits du gestionnaire de files d'attente.
3. Assurez-vous que le profil utilisateur QMQM possède le même mot de passe sur les trois serveurs.

4. Installez IBM i NetServer; voir [i5/OS NetServer](#).
5. Créez des procédures pour transférer l'ASP indépendant du gestionnaire de files d'attente défaillant vers le serveur de secours qui prend le relais. Certaines des techniques décrites dans *SupportPac MC41: Configuration de IBM MQ for iSeries for High Availability* peuvent être utiles lors de la conception de vos procédures de transfert ASP indépendant.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour créer la configuration illustrée dans la [Figure 40](#), à la page 323. Les données du gestionnaire de files d'attente sont connectées à l'aide d' IBM i NetServer.

- Créez des connexions depuis ALPHA et BETA vers le partage de répertoire sur GAMMA qui permet de stocker les données du gestionnaire de files d'attente. La tâche configure également les droits d'accès, les profils utilisateur et les mots de passe nécessaires.
- Créez le gestionnaire de files d'attente QM1 sur le serveur IBM i , ALPHA.
- Ajoutez les informations de contrôle du gestionnaire de files d'attente pour QM1 sur l'autre serveur IBM i , BETA.
- Démarrez le gestionnaire de files d'attente en autorisant une instance de secours.

Procédure

1. Exécutez la tâche [«Création d'un partage de réseau pour les données de gestionnaire de files d'attente à l'aide de NetServer sous IBM i»](#), à la page 297.

Par conséquent, ALPHA et BETA ont un partage, /QNTC/GAMMA/WMQ, qui pointe vers /QIBM/UserData/mqm/qmgrs sur GAMMA. Les profils utilisateur QMQM et QMQMADM disposent des droits nécessaires et QMQM possède des mots de passe correspondants sur les trois systèmes.

2. Créez le gestionnaire de files d'attente QM1 sur ALPHA, en sauvegardant les données du gestionnaire de files d'attente sur GAMMA.

```
CRTMQM MQMNAME(QM1) UDLMSGQ(SYSTEM.DEAD.LETTER.QUEUE)
MQMDIR(' /QNTC/GAMMA/WMQ ')
```

Le chemin d'accès utilise NetServer pour créer les données du gestionnaire de files d'attente.

3. Utilisez la commande, pour inspecter les données de configuration IBM MQ créées par pour sur ALPHA.

Les informations sont nécessaires à l'étape suivante.

Dans cet exemple, la configuration suivante est créée dans ALPHA pour:

```
Name=QM1
Prefix=/QIBM/UserData/mqm
Library=QMOM1
Directory=QM1
DataPath= /QNTC/GAMMA/WMQ /QM1
```

4. Créez une instance de gestionnaire de files d'attente de QM1 sur BETA à l'aide de la commande. Exécutez la commande suivante sur BETA pour modifier les informations de contrôle du gestionnaire de files d'attente sur BETA.

```
ADDMQMINF MQMNAME(QM1)
PREFIX('/QIBM/UserData/mqm')
MQMDIR(QM1)
MQMLIB(QMOM1)
DATAPATH('/QNTC/GAMMA/WMQ /QM1 ')
```

Conseil : Copiez et collez les informations de configuration. La section du gestionnaire de files d'attente est la même sur ALPHA et BETA.

5. Démarrez les instances de gestionnaire de files d'attente sur ALPHA et BETA.

- a) Démarrez la première instance sur ALPHA, ce qui en fait l'instance active. Activation du basculement vers une instance de secours.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- b) Démarrez la deuxième instance sur BETA, ce qui en fait l'instance de secours.

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

Résultats

Permet de vérifier le statut du gestionnaire de files d'attente:

1. Le statut de l'instance de gestionnaire de files d'attente sur ALPHA doit être.
2. Le statut de l'instance de gestionnaire de files d'attente sur BETA doit être.

Exemple

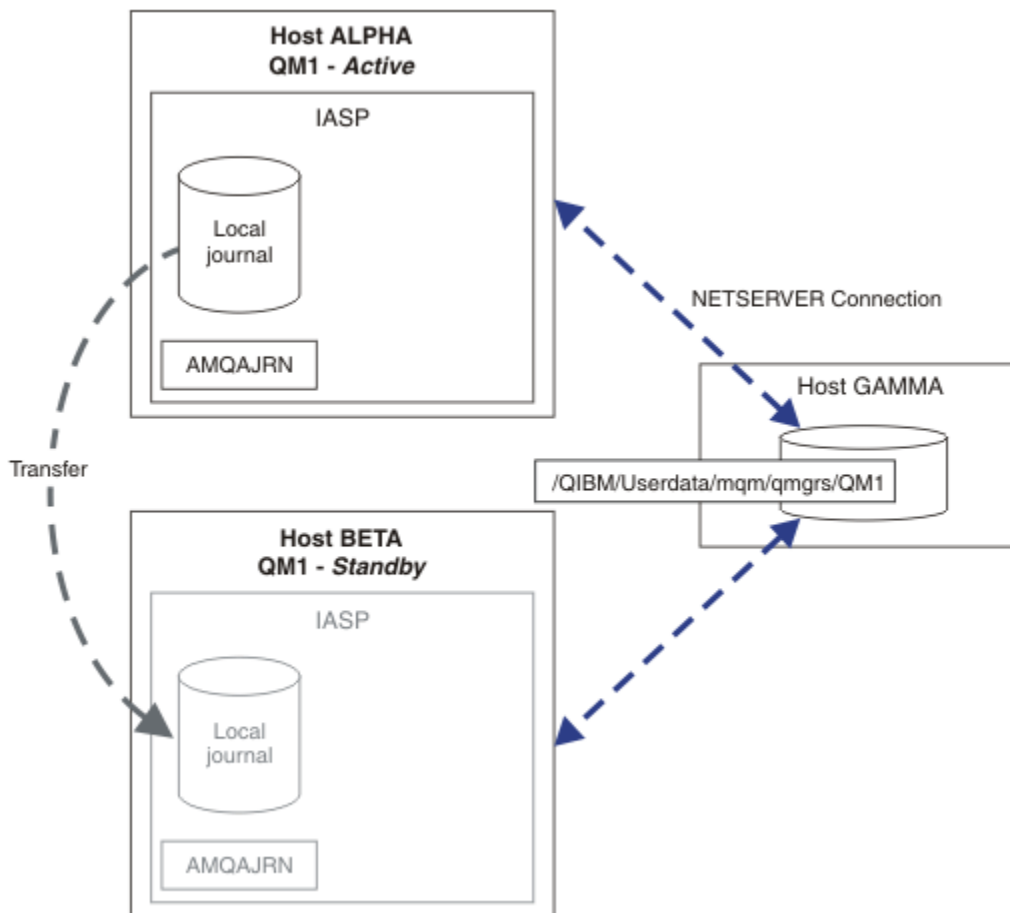


Figure 40. Transfert d'un journal de gestionnaire de files d'attente à l'aide d'un ASP indépendant

Que faire ensuite

- Vérifiez que les instances actives et de secours basculent automatiquement. Vous pouvez exécuter les exemples de programmes à haute disponibilité pour tester le basculement ; voir [Exemples de](#)

programmes à haute disponibilité. Les exemples de programme sont des clients 'C'. Vous pouvez les exécuter à partir d'une plateforme Windows ou Unix.

1. Démarrez les exemples de programmes à haute disponibilité.
2. Sous ALPHA, arrêtez le gestionnaire de files d'attente demandant le basculement:

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) ALSWITCH(*YES)
```

3. Vérifiez que l'instance de sur BETA est active.
4. Redémarrer sur ALPHA

```
STRMQM MQMNAME(QM1) STANDBY(*YES)
```

- Examinez les autres configurations à haute disponibilité:
 1. Utilisez NetServer pour placer les données du gestionnaire de files d'attente sur un serveur IBM i .
 2. Au lieu d'utiliser un ASP indépendant pour transférer le journal du gestionnaire de files d'attente sur le serveur de secours, utilisez la journalisation à distance pour reproduire le journal sur le serveur de secours.

IBM i ASP indépendants et haute disponibilité

Les ASP indépendants permettent de déplacer des applications et des données entre les serveurs. La flexibilité des ASP indépendants signifie qu'ils constituent la base de certaines solutions à haute disponibilité IBM i . Lorsque vous envisagez d'utiliser un ASP ou un ASP indépendant pour le journal du gestionnaire de files d'attente, vous devez prendre en compte une autre configuration à haute disponibilité basée sur des ASP indépendants.

Les pools de mémoire secondaire (ASP) sont un bloc de construction de l'architecture IBM i . Les unités de disques sont regroupées pour former un ASP unique. En plaçant des objets dans différents ASP, vous pouvez protéger les données d'un ASP contre les défaillances de disque dans un autre ASP.

Chaque serveur IBM i possède au moins un ASP *de base* , appelé ASP système. Il est désigné comme ASP1 et parfois appelé *SYSBAS. Vous pouvez configurer jusqu'à 31 ASP *utilisateur* de base supplémentaires qui ne peuvent pas être distingués de l'ASP système du point de vue de l'application, car ils partagent le même espace de nom. En utilisant plusieurs ASP de base pour distribuer des applications sur de nombreux disques, vous pouvez améliorer les performances et réduire le temps de reprise. L'utilisation de plusieurs ASP de base peut également fournir un certain degré d'isolement en cas de panne de disque, mais elle n'améliore pas la fiabilité globale.

Les ASP indépendants sont un type spécial d'ASP. Ils sont souvent appelés pools de stockage sur disque indépendants. Les pools de stockage sur disque indépendants sont des composants clés de la haute disponibilité IBM i . Vous pouvez stocker des données et des applications qui se considèrent comme indépendantes du système en cours auquel elles sont connectées sur des unités de stockage sur disque indépendantes. Vous pouvez configurer des ASP indépendants commutables ou non commutables. Du point de vue de la disponibilité, vous ne vous souciez généralement que des ASP indépendants commutables, qui peuvent être transférés automatiquement d'un serveur à un autre. Par conséquent, vous pouvez déplacer les applications et les données de l'ASP indépendant d'un serveur à un autre.

Contrairement aux ASP utilisateur de base, les ASP indépendants ne partagent pas le même espace de nom que l'ASP système. Les applications qui utilisent des ASP utilisateur nécessitent des modifications pour fonctionner avec un ASP indépendant. Vous devez vérifier que votre logiciel et le logiciel tiers que vous utilisez fonctionnent dans un environnement ASP indépendant.

Lorsque l'ASP indépendant est connecté à un autre serveur, l'espace de nom de l'ASP indépendant doit être combiné avec l'espace de nom de l'ASP système. Ce processus est appelé *mise en fonction* de l'ASP indépendant. Vous pouvez mettre en fonction un ASP indépendant sans effectuer d'IPL du serveur. La prise en charge de la mise en grappe est requise pour le transfert automatique des ASP indépendants d'un serveur à un autre.

Création de solutions fiables avec des ASP indépendants

La journalisation dans un ASP indépendant, plutôt que dans un ASP et à l'aide de la réplication de journal, offre un autre moyen de fournir au gestionnaire de files d'attente de secours une copie du journal local à partir de l'instance de gestionnaire de files d'attente défaillante. Pour transférer automatiquement l'ASP indépendant vers un autre serveur, vous devez avoir installé et configuré la prise en charge de la mise en grappe. Il existe un certain nombre de solutions à haute disponibilité pour les ASP indépendants en fonction de la prise en charge du cluster et de la mise en miroir de disque de bas niveau, que vous pouvez combiner avec des gestionnaires de files d'attente multi-instance ou les remplacer.

La liste suivante décrit les composants nécessaires à la génération d'une solution fiable basée sur des ASP indépendants.

Journalisation

Les gestionnaires de files d'attente et d'autres applications utilisent des journaux locaux pour écrire des données persistantes en toute sécurité sur le disque afin de les protéger contre la perte de données en mémoire en cas de défaillance du serveur. Cette cohérence est parfois appelée cohérence ponctuelle. Elle ne garantit pas la cohérence des mises à jour multiples qui ont lieu sur une période donnée.

Commitment Control

En utilisant des transactions globales, vous pouvez coordonner les mises à jour des messages et des bases de données afin que les données écrites dans le journal soient cohérentes. Il assure la cohérence sur une période de temps en utilisant un protocole de validation en deux phases.

Disque commuté

Les disques commutés sont gérés par le groupe de ressources en grappe d'unité dans une grappe à haute disponibilité. Le groupe de ressources en grappe bascule automatiquement les ASP indépendants vers un nouveau serveur en cas d'indisponibilité non planifiée. Les groupes de ressources en grappe sont géographiquement limités à l'étendue du bus d'E-S local.

En configurant votre journal local sur un ASP indépendant commutable, vous pouvez transférer le journal sur un autre serveur et reprendre le traitement des messages. Aucune modification apportée aux messages persistants sans contrôle de point de synchronisation ou validés avec contrôle de point de synchronisation n'est perdue, sauf si l'ASP indépendant est défaillant.

Si vous utilisez à la fois la journalisation et le contrôle de validation sur des ASP indépendants commutables, vous pouvez transférer des journaux de base de données et des journaux de gestionnaire de files d'attente vers un autre serveur et reprendre le traitement des transactions sans perte de cohérence ni de transactions validées.

Mise en miroir intersite (XSM)

XSM met en miroir l'ASP indépendant principal vers un ASP indépendant secondaire géographiquement éloigné sur un réseau TCP/IP et transfère automatiquement le contrôle en cas de défaillance. Vous avez le choix entre la configuration d'un miroir synchrone ou asynchrone. La mise en miroir synchrone réduit les performances du gestionnaire de files d'attente car les données sont mises en miroir avant la fin des opérations d'écriture sur le système de production, mais elle garantit que l'ASP indépendant secondaire est à jour. Si vous utilisez la mise en miroir asynchrone, vous ne pouvez pas garantir que l'ASP indépendant secondaire est à jour. La mise en miroir asynchrone assure la cohérence de l'ASP indépendant secondaire.

Il existe trois technologies XSM.

Protection géographique par disque miroir

La protection géographique par disque miroir est une extension de la mise en grappe, qui vous permet de basculer entre les ASP indépendants sur une vaste zone. Il possède à la fois des modes synchrone et asynchrone. Vous pouvez garantir la haute disponibilité uniquement en mode synchrone, mais la séparation des ASP indépendants peut avoir un impact trop important sur les performances. Vous pouvez combiner la protection géographique par disque miroir avec un disque commuté pour assurer la haute disponibilité en local et la reprise après incident à distance.

Mise en miroir Metro

La mise en miroir Metro est un service de niveau unité qui fournit une mise en miroir synchrone locale rapide sur des distances plus longues que le bus local. Vous pouvez le combiner avec un gestionnaire de files d'attente multi-instance pour vous assurer de la haute disponibilité du gestionnaire de files d'attente et en disposant de deux copies de l'ASP indépendant, de la haute disponibilité du journal du gestionnaire de files d'attente.

Mise en miroir globale

La mise en miroir globale est un service de niveau unité qui fournit une mise en miroir asynchrone et qui convient à la sauvegarde et à la reprise après incident sur des distances plus longues, mais qui n'est pas un choix normal pour la haute disponibilité, car il ne maintient que la cohérence du point de cohérence plutôt que la devise.

Les principaux points de décision que vous devez prendre en considération sont les suivants:

ASP ou ASP indépendant?

Il n'est pas nécessaire d'exécuter un cluster à haute disponibilité IBM i pour utiliser des gestionnaires de files d'attente multi-instance. Vous pouvez choisir des ASP indépendants, si vous utilisez déjà des ASP indépendants, ou si vous avez des exigences de disponibilité pour d'autres applications qui requièrent des ASP indépendants. Il peut être utile de combiner des ASP indépendants avec des gestionnaires de files d'attente multi-instance pour remplacer la surveillance des gestionnaires de files d'attente afin de détecter les incidents de gestionnaire de files d'attente.

Disponibilité?

Quel est l'objectif de temps de reprise (RTO)? Si vous avez besoin de l'apparence d'un comportement quasi ininterrompu, quelle solution a le temps de récupération le plus rapide?

Disponibilité du journal?

Comment éliminer le journal en tant que point de défaillance unique. Vous pouvez adopter une solution matérielle, à l'aide de périphériques RAID 1 ou d'une version supérieure, ou vous pouvez combiner ou utiliser une solution logicielle à l'aide de journaux de réplique ou de la mise en miroir de disque.

Distance?

A quelle distance se trouvent les instances de gestionnaire de files d'attente actives et de secours. Vos utilisateurs peuvent-ils tolérer la dégradation des performances de la réplication synchrone sur des distances supérieures à environ 250 mètres?

Compétences?

Il y a du travail à faire pour automatiser les tâches administratives liées à la gestion et à l'exercice réguliers de la solution. Les compétences requises pour effectuer l'automatisation sont différentes pour les solutions basées sur les ASP et les ASP indépendants.



Suppression d'un gestionnaire de files d'attente multi-instance sous IBM i

Avant de supprimer un gestionnaire de files d'attente multi-instance, arrêtez la journalisation à distance et supprimez des instances de gestionnaire de files d'attente.

Avant de commencer

1. Dans cet exemple, deux instances du gestionnaire de files d'attente QM1 sont définies sur les serveurs ALPHA et BETA. ALPHA est l'instance active et BETA est l'instance de secours. Les données du gestionnaire de files d'attente associées au gestionnaire de files d'attente QM1 sont stockées sur le serveur IBM i GAMMA, à l'aide de NetServer. Voir [«Création d'un gestionnaire de files d'attente multi-instance à l'aide de la mise en miroir du journal et de NetServer sous IBM i»](#), à la page 311.
2. ALPHA et BETA doivent être connectés pour que les journaux éloignés définis puissent être supprimés par IBM MQ.
3. Vérifiez que le répertoire /QNTC et le partage de fichiers du répertoire du serveur sont accessibles à l'aide des commandes système **EDTF** ou **WRKLNK**

Pourquoi et quand exécuter cette tâche

Avant de supprimer un gestionnaire de files d'attente multi-instance d'un serveur à l'aide de la commande **DLTMQM**, supprimez toutes les instances de gestionnaire de files d'attente sur d'autres serveurs à l'aide de la commande **RMVMQMINF**.

Lorsque vous supprimez une instance de gestionnaire de files d'attente à l'aide de la commande **RMVMQMINF**, les journaux locaux et distants préfixés par AMQet associés à l'instance sont supprimés. Les informations de configuration relatives à l'instance de gestionnaire de files d'attente, locale sur le serveur, sont également supprimées.

N'exécutez pas la commande **RMVMQMINF** sur le serveur contenant l'instance restante du gestionnaire de files d'attente. Cela empêche **DLTMQM** de fonctionner correctement.

Supprimez le gestionnaire de files d'attente à l'aide de la commande **DLTMQM**. Les données du gestionnaire de files d'attente sont supprimées du partage de réseau. Les journaux locaux et éloignés préfixés avec AMQ et associés à l'instance sont supprimés. **DLTMQM** supprime également les informations de configuration relatives à l'instance de gestionnaire de files d'attente, locale sur le serveur.

Dans l'exemple, il n'y a que deux instances de gestionnaire de files d'attente. IBM MQ prend en charge une configuration multi-instance en cours d'exécution qui comporte une instance de gestionnaire de files d'attente active et une instance de secours. Si vous avez créé des instances de gestionnaire de files d'attente supplémentaires à utiliser dans des configurations en cours d'exécution, supprimez-les à l'aide de la commande **RMVMQMINF** avant de supprimer l'instance restante.

Procédure

1. Exécutez la commande **CHGMQMJRN RMTJRNSTS (*INACTIVE)** sur chaque serveur pour rendre la journalisation à distance entre les instances de gestionnaire de files d'attente inactive.

a) Sur ALPHA:

```
CHGMQMJRN MQMNAME('QM1')
RMTJRNRDB('BETA') RMTJRNSTS(*INACTIVE)
```

b) Sur BETA:

```
CHGMQMJRN MQMNAME('QM1')
RMTJRNRDB('ALPHA') RMTJRNSTS(*INACTIVE)
```

2. Exécutez la commande **ENDMQM** sur ALPHA, l'instance de gestionnaire de files d'attente active, pour arrêter les deux instances de QM1.

```
ENDMQM MQMNAME(QM1) OPTION(*IMMED) INSTANCE(*ALL) ENDCCTJOB(*YES)
```

3. Exécutez la commande **RMVMQMINF** sur ALPHA pour supprimer les ressources de gestionnaire de files d'attente de l'instance de ALPHA et de BETA.

```
RMVMQMINF MQMNAME(QM1)
```

RMVMQMINF supprime les informations de configuration du gestionnaire de files d'attente pour QM1 de ALPHA. Si le nom de journal est préfixé par AMQ, il supprime le journal local associé à QM1 de ALPHA. Si le nom de journal est préfixé par AMQ et qu'un journal éloigné a été créé, il supprime également le journal éloigné de BETA.

4. Exécutez la commande **DLTMQM** sur BETA pour supprimer QM1.

```
DLTMQM MQMNAME(QM1)
```

DLTMQM supprime les données du gestionnaire de files d'attente du partage de réseau sur GAMMA. Elle supprime les informations de configuration du gestionnaire de files d'attente pour QM1 de BETA.

Si le nom du journal est précédé du préfixe AMQ, le journal local associé à QM1 est supprimé de BETA. Si le nom du journal est préfixé par AMQ et qu'un journal éloigné a été créé, il supprime également le journal éloigné de ALPHA.

Résultats

DLTMQM et **RMVMQMINF** suppriment les journaux locaux et éloignés créés par **CRTMQM** et **ADDMQJRN**. Les commandes suppriment également les récepteurs de journal. Les journaux et les récepteurs de journal doivent respecter la convention de dénomination selon laquelle les noms doivent commencer par AMQ. **DLTMQM** et **RMVMQMINF** suppriment les objets du gestionnaire de files d'attente, les données du gestionnaire de files d'attente et les informations de configuration du gestionnaire de files d'attente de `mqs.ini`.

Que faire ensuite

Une autre approche consiste à émettre les commandes suivantes après la désactivation de la journalisation à l'étape «1», à la page 327 et avant l'arrêt des instances de gestionnaire de files d'attente. Sinon, si vous n'avez pas suivi la convention de dénomination, vous devez supprimer les journaux et les récepteurs de journal par leur nom.

1. Sur ALPHA:

```
RMVMQMJRN MQMNAME('QM1') RMTJRNRDB('BETA')
```

2. Sur BETA:

```
RMVMQMJRN MQMNAME('QM1') RMTJRNRDB('ALPHA')
```

Après avoir supprimé les journaux, passez au reste des étapes.

Sauvegarde d'un gestionnaire de files d'attente multi-instance sous IBM i

Cette procédure explique comment sauvegarder les objets de gestionnaire de files d'attente sur le serveur local et les données du gestionnaire de files d'attente sur le serveur de fichiers réseau. Adaptez l'exemple pour sauvegarder des données pour d'autres gestionnaires de files d'attente.

Avant de commencer

Dans cet exemple, les données de gestionnaire de files d'attente associées au gestionnaire de files d'attente QM1 sont stockées sur le serveur IBM i appelé GAMMA, à l'aide de NetServer. Voir «Création d'un gestionnaire de files d'attente multi-instance à l'aide de la mise en miroir du journal et de NetServer sous IBM i», à la page 311. IBM MQ est installé sur les serveurs, ALPHA et BETA. Le gestionnaire de files d'attente, QM1, est configuré sur ALPHA et BETA.

Pourquoi et quand exécuter cette tâche

IBM i ne prend pas en charge la sauvegarde des données à partir d'un répertoire distant. Sauvegardez les données du gestionnaire de files d'attente sur un système de fichiers distant à l'aide des procédures de sauvegarde locales sur le serveur de système de fichiers. Dans cette tâche, le système de fichiers réseau se trouve sur un serveur IBM i, GAMMA. Les données du gestionnaire de files d'attente sont sauvegardées dans un fichier sauvegarde sur GAMMA.

Si le système de fichiers réseau se trouve sous Windows ou Linux, vous pouvez stocker les données du gestionnaire de files d'attente dans un fichier compressé, puis les sauvegarder. Si vous disposez d'un système de sauvegarde, tel que Tivoli Storage Manager, utilisez-le pour sauvegarder les données du gestionnaire de files d'attente.

Procédure

1. Créez un fichier sauvegarde sur ALPHA pour la bibliothèque de gestionnaire de files d'attente associée à QM1.

Utilisez le nom de la bibliothèque du gestionnaire de files d'attente pour nommer le fichier sauvegarde.

```
CRTSAVF FILE(QGPL/QMQM1)
```

2. Sauvegardez la bibliothèque du gestionnaire de files d'attente dans le fichier sauvegarde sur ALPHA.

```
SAVLIB LIB(QMQM1) DEV(*SAVF) SAVF(QGPL/QMQM1)
```

3. Créez un fichier sauvegarde pour le répertoire de données du gestionnaire de files d'attente sur GAMMA.

Utilisez le nom du gestionnaire de files d'attente pour nommer le fichier sauvegarde.

```
CRTSAVF FILE(QGPL/QMDQM1)
```

4. Sauvegardez la copie des données du gestionnaire de files d'attente à partir du répertoire local sur GAMMA.

```
SAV DEV('/QSYS.LIB/QGPL.LIB/QMDQM1.FILE') OBJ('/QIBM/Userdata/mqm/qmgrs/QM1')
```



Commandes de configuration de gestionnaires de files d'attente multi-instance

IBM MQ comporte des commandes permettant de simplifier la configuration de la réplication de journal, d'ajouter de nouvelles instances de gestionnaire de files d'attente et de configurer les gestionnaires de files d'attente pour utiliser un ASP indépendant.

Les commandes de journal permettant de créer et de gérer des journaux locaux et éloignés sont les suivantes:

ADDMQMJRN

Avec cette commande, vous pouvez créer des journaux locaux et éloignés nommés pour une instance de gestionnaire de files d'attente et configurer si la réplication est synchrone ou asynchrone, en quoi consiste le délai d'attente synchrone et si le journal éloigné doit être activé immédiatement.

CHGMQMJRN

La commande modifie les paramètres de délai d'attente, de statut et de distribution affectant les journaux de réplique.

RMVMQMJRN

Supprime les journaux *distant*s nommés d'une instance de gestionnaire de files d'attente.

WRKMQMJRN

Répertorie l'état des journaux locaux et éloignés pour une instance de gestionnaire de files d'attente locale.

Ajoutez et gérez des instances de gestionnaire de files d'attente supplémentaires à l'aide des commandes suivantes, qui modifient le fichier `mqs.ini`.

ADDMQMINF

La commande utilise les informations que vous avez extraites du fichier `mqs.ini` à l'aide de la commande `DSPMQMINF` pour ajouter une nouvelle instance de gestionnaire de files d'attente sur un autre serveur IBM i.

RMVMQMINF

Supprimez une instance de gestionnaire de files d'attente. Utilisez cette commande pour supprimer une instance d'un gestionnaire de files d'attente existant ou pour supprimer les informations de configuration d'un gestionnaire de files d'attente qui a été supprimé d'un autre serveur.

La commande **CRTMQM** comporte trois paramètres pour vous aider à configurer un gestionnaire de files d'attente multi-instance.

MQMDIRP (*DFT | *préfixe-répertoire*)

Utilisez ce paramètre pour sélectionner un point de montage mappé aux données du gestionnaire de files d'attente sur le stockage réseau.

ASP (*SYSTEM|*ASPDEV| *numéro-pool-mémoire-secondaire*)

Indiquez *SYSTEM ou un *numéro de pool de mémoire secondaire* pour placer le journal du gestionnaire de files d'attente sur le système ou sur un ASP utilisateur de base. Sélectionnez l'option *ASPDEV et définissez un nom d'unité à l'aide du paramètre **ASPDEV** pour placer le journal du gestionnaire de files d'attente sur un ASP indépendant.

ASPDEV (*ASP|*nom-unité*)

Indiquez le *nom d'unité* d'une unité ASP indépendante principale ou secondaire. La sélection de *ASP a le même résultat que la spécification de **ASP** (*SYSTEM).

IBM i Remarques sur les performances et la reprise sur disque sous IBM i

Utilisez différents pools de mémoire secondaire pour améliorer les performances et la fiabilité.

Si vous utilisez un grand nombre de messages persistants ou de messages volumineux dans vos applications, le temps passé à écrire ces messages sur le disque devient un facteur important dans les performances du système.

Vérifiez que vous disposez d'une activation de disque suffisante pour faire face à cette possibilité ou envisagez un pool de mémoire secondaire (ASP) distinct dans lequel placer les récepteurs de journal de votre gestionnaire de files d'attente.

Vous pouvez spécifier l'ASP dans lequel la bibliothèque de votre gestionnaire de files d'attente et les journaux sont stockés lorsque vous créez votre gestionnaire de files d'attente à l'aide du paramètre ASP de **CRTMQM**. Par défaut, la bibliothèque du gestionnaire de files d'attente, les journaux et les données IFS sont stockés dans l'ASP système.

Les ASP permettent d'isoler des objets sur une ou plusieurs unités de disque spécifiques. Cela peut également réduire la perte de données en raison d'une défaillance de support de disque. Dans la plupart des cas, seules les données stockées sur les unités de disque de l'ASP concerné sont perdues.

Il est recommandé de stocker la bibliothèque du gestionnaire de files d'attente et les données de journal dans des ASP utilisateur distincts du système de fichiers IFS racine afin de fournir une reprise en ligne et de réduire les conflits d'accès au disque.

Pour plus d'informations, voir [Sauvegarde et reprise](#).

IBM i Utilisation de SAVLIB pour sauvegarder les bibliothèques IBM MQ sous IBM i

Vous ne pouvez pas utiliser SAVLIB LIB(*ALLUSR) pour sauvegarder les bibliothèques IBM MQ, car ces bibliothèques ont des noms commençant par Q.

Vous pouvez utiliser SAVLIB LIB(QM*) pour sauvegarder toutes les bibliothèques du gestionnaire de files d'attente, mais uniquement si vous utilisez une unité de sauvegarde autre que *SAVF. Pour DEV(*SAVF), vous devez utiliser une commande SAVLIB pour chaque bibliothèque de gestionnaire de files d'attente de votre système.

IBM i mise au repos IBM MQ for IBM i

Cette section explique comment mettre au repos (arrêt approprié) IBM MQ for IBM i.

Pour mettre au repos IBM MQ for IBM i:

1. Connectez-vous à une nouvelle session IBM MQ for IBM i interactive en vous assurant que vous n'accédez à aucun objet.

2. Assurez-vous que vous disposez :

- Droit *ALLOBJ ou droit de gestion d'objet pour la bibliothèque QMQM
- Droits suffisants pour utiliser la commande ENDSBS

3. Informez tous les utilisateurs que vous allez arrêter IBM MQ for IBM i.

4. La procédure à suivre varie selon que vous souhaitez arrêter (mettre au repos) un seul gestionnaire de files d'attente (où d'autres gestionnaires de files d'attente peuvent exister) (voir «[Arrêt d'un gestionnaire de files d'attente unique pour IBM MQ for IBM i](#)», à la page 331). ou tous les gestionnaires de files d'attente (voir «[Arrêt de tous les gestionnaires de files d'attente pour IBM MQ for IBM i](#)», à la page 333).

ENDMQM, paramètre ENDCCTJOB (*YES)

Le paramètre ENDMQM ENDCCTJOB (*YES) fonctionne différemment dans IBM MQ for IBM i 6.0 et versions ultérieures par rapport aux versions précédentes.

Dans les versions précédentes, lorsque vous spécifiez ENDCCTJOB (*YES), MQ force l'arrêt de vos applications pour vous.

Sous IBM MQ for IBM i 6.0 ou version ultérieure, lorsque vous spécifiez ENDCCTJOB (*YES), vos applications ne sont pas arrêtées mais sont déconnectées du gestionnaire de files d'attente.

Si vous spécifiez ENDCCTJOB (*YES) et que vous avez des applications qui ne sont pas écrites pour détecter qu'un gestionnaire de files d'attente est en cours d'arrêt, lors de la prochaine émission d'un nouvel appel MQI, l'appel sera renvoyé avec une erreur MQRC_CONNECTION_BROKEN (2009).

Au lieu d'utiliser ENDCCTJOB (*YES), utilisez le paramètre ENDCCTJOB (*NO) et l'option 22 de la commande WRKMQM (Gestion des travaux) pour arrêter manuellement les travaux d'application qui empêchent le redémarrage d'un gestionnaire de files d'attente.

IBM i Arrêt d'un gestionnaire de files d'attente unique pour IBM MQ for IBM i

Utilisez ces informations pour comprendre les trois types d'arrêt.

Dans les procédures qui suivent, nous utilisons un exemple de nom de gestionnaire de files d'attente QMgr1 et un exemple de nom de sous-système SUBX. Si nécessaire, remplacez ces noms par vos propres valeurs.

Arrêt planifié

Arrêt planifié d'un gestionnaire de files d'attente sous IBM i

1. Avant l'arrêt, exécutez:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMgr1) DSPJRNTA(*YES)
```

2. Pour arrêter le gestionnaire de files d'attente, exécutez:

```
ENDMQM MQMNAME(QMgr1) OPTION(*CNTRLD)
```

Si QMgr1 ne s'arrête pas, le canal ou les applications sont probablement occupés.

3. Si vous devez arrêter QMgr1 immédiatement, exécutez la commande suivante:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

Arrêt non planifié

1. Pour arrêter le gestionnaire de files d'attente, exécutez:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Si QMgr1 ne s'arrête pas, le canal ou les applications sont probablement occupés.

2. Si vous devez arrêter QMgr1 immédiatement, exécutez ce qui suit:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(15)
```

Arrêt en cas de conditions anormales

1. Pour arrêter le gestionnaire de files d'attente, exécutez:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Si QMgr1 ne s'arrête pas, passez à l'étape 3 en indiquant que:

- QMgr1 se trouve dans son propre sous-système, ou
 - Vous pouvez arrêter tous les gestionnaires de files d'attente qui partagent le même sous-système que QMgr1. Utilisez la procédure d'arrêt non planifié pour tous les gestionnaires de files d'attente de ce type.
2. Une fois que vous avez effectué toutes les étapes de la procédure pour tous les gestionnaires de files d'attente partageant le sous-système (SUBX dans nos exemples), exécutez:

```
ENDSBS SUBX *IMMED
```

Si cette commande échoue, arrêtez tous les gestionnaires de files d'attente à l'aide de la procédure d'arrêt non planifiée et effectuez un IPL sur votre machine.

Avertissement : N'utilisez pas les travaux ENDJOBABN for IBM MQ qui ne parviennent pas à s'arrêter suite à ENDJOB ou ENDSBS, sauf si vous êtes prêt à effectuer un IPL sur votre machine immédiatement après.

3. Démarrez le sous-système en exécutant:

```
STRSBS SUBX
```

4. Arrêtez immédiatement le gestionnaire de files d'attente en exécutant:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)  
ENDCCTJOB(*YES) TIMEOUT(10)
```

5. Redémarrez le gestionnaire de files d'attente en exécutant:

```
STRMQM MQMNAME(QMgr1)
```

Si cela échoue, et que vous:

- Vous avez redémarré votre machine en exécutant un IPL, ou
- N'avoir qu'un seul gestionnaire de files d'attente

Rangez la mémoire partagée IBM MQ en exécutant:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

avant de répéter l'étape 5.

Si le redémarrage du gestionnaire de files d'attente prend plus de quelques secondes, IBM MQ ajoute des messages d'état par intermittence au journal des travaux détaillant la progression du démarrage.

Si vous rencontrez toujours des problèmes lors du redémarrage de votre gestionnaire de files d'attente, contactez le support IBM . Toute autre action que vous pourriez effectuer pourrait endommager le gestionnaire de files d'attente, ce qui ne permet pas à IBM MQ d'effectuer de reprise.

Arrêt de tous les gestionnaires de files d'attente pour IBM MQ for IBM i

Utilisez ces informations pour comprendre les trois types d'arrêt.

Les procédures sont presque les mêmes que pour un gestionnaire de files d'attente unique, mais en utilisant *ALL à la place du nom du gestionnaire de files d'attente lorsque cela est possible, et en utilisant une commande à plusieurs reprises en utilisant chaque nom de gestionnaire de files d'attente à tour de rôle. Tout au long des procédures, nous utilisons un exemple de nom de gestionnaire de files d'attente QMgr1 et un exemple de nom de sous-système SUBX. Remplacez-les par les vôtres.

Arrêt planifié

1. Une heure avant l'arrêt, exécutez:

```
RCDMQMIMG OBJ(*ALL) OBJTYPE(*ALL) MQMNAME(QMgr1) DSPJRNTA(*YES)
```

Répétez cette opération pour chaque gestionnaire de files d'attente que vous souhaitez arrêter.

2. Pour arrêter le gestionnaire de files d'attente, exécutez:

```
ENDMQM MQMNAME(QMgr1) OPTION(*CNTRLD)
```

Répétez cette opération pour chaque gestionnaire de files d'attente que vous souhaitez arrêter ; des commandes distinctes peuvent être exécutées en parallèle.

Si un gestionnaire de files d'attente ne s'arrête pas dans un délai raisonnable (par exemple, 10 minutes), passez à l'étape 3.

3. Pour arrêter immédiatement tous les gestionnaires de files d'attente, exécutez la commande suivante:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

Arrêt non planifié

1. Pour arrêter un gestionnaire de files d'attente, exécutez:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Répétez cette opération pour chaque gestionnaire de files d'attente que vous souhaitez arrêter ; des commandes distinctes peuvent être exécutées en parallèle.

Si les gestionnaires de files d'attente ne s'arrêtent pas, le canal ou les applications sont probablement occupés.

2. Si vous devez arrêter les gestionnaires de files d'attente immédiatement, exécutez la commande suivante:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

Arrêt en cas de conditions anormales

1. Pour arrêter les gestionnaires de files d'attente, exécutez:

```
ENDMQM MQMNAME(QMgr1) OPTION(*IMMED)
```

Répétez cette opération pour chaque gestionnaire de files d'attente que vous souhaitez arrêter ; des commandes distinctes peuvent être exécutées en parallèle.

2. Arrêtez les sous-systèmes (SUBX dans nos exemples) en exécutant:

```
ENDSBS SUBX *IMMED
```

Répétez cette opération pour chaque sous-système que vous souhaitez arrêter ; des commandes distinctes peuvent être exécutées en parallèle.

Si cette commande échoue, effectuez un IPL sur votre système.

Avertissement : N'utilisez pas ENDJOBABN pour les travaux qui ne parviennent pas à s'arrêter suite à ENDJOB ou ENDSBS, sauf si vous êtes prêt à effectuer un IPL sur votre système immédiatement après.

3. Démarrez les sous-systèmes en exécutant:

```
STRSBS SUBX
```

Répétez cette opération pour chaque sous-système que vous souhaitez démarrer.

4. Arrêtez immédiatement les gestionnaires de files d'attente en exécutant:

```
ENDMQM MQMNAME(*ALL) OPTION(*IMMED)
ENDCCTJOB(*YES) TIMEOUT(15)
```

5. Redémarrez les gestionnaires de files d'attente en exécutant:

```
STRMQM MQMNAME(QMgr1)
```

Répétez cette opération pour chaque gestionnaire de files d'attente que vous souhaitez démarrer.

Si le redémarrage d'un gestionnaire de files d'attente prend plus de quelques secondes, IBM MQ affiche des messages d'état détaillant par intermittence la progression du démarrage.

Si vous rencontrez toujours des problèmes lors du redémarrage d'un gestionnaire de files d'attente, contactez le support IBM . Toute autre action que vous pourriez effectuer pourrait endommager les gestionnaires de files d'attente, laissant MQSeries ou IBM MQ dans l'impossibilité d'effectuer une reprise.

L'administration des gestionnaires de files d'attente et des ressources associées inclut les tâches que vous effectuez fréquemment pour activer et gérer ces ressources. Choisissez la méthode que vous préférez pour administrer vos gestionnaires de files d'attente et les ressources associées.

IBM MQ for z/OS peut être contrôlé et géré par un ensemble d'utilitaires et de programmes fournis avec le produit. Vous pouvez utiliser les commandes IBM MQ Script (MQSC) ou Programmable Command Formats

(PCF) pour administrer IBM MQ for z/OS. Pour plus d'informations sur l'utilisation des commandes pour IBM MQ for z/OS, voir [«Emission de commandes sur IBM MQ for z/OS»](#), à la page 335.

IBM MQ for z/OS fournit également un ensemble de programmes utilitaires pour vous aider dans l'administration du système. Pour plus d'informations sur les différents programmes utilitaires et sur leur utilisation, voir [«Les utilitaires IBM MQ for z/OS»](#), à la page 344.

Pour plus de détails sur la façon d'administrer IBM MQ for z/OS et les différentes tâches d'administration que vous pouvez avoir à effectuer, voir les liens suivants:

Concepts associés

[«Administration d'IBM MQ»](#), à la page 5

Pour administrer vos gestionnaires de files d'attente IBM MQ et les ressources associées, choisissez la méthode de votre choix parmi un ensemble de tâches que vous pouvez utiliser pour activer et gérer ces ressources.

Tâches associées

[«Administration des objets IBM MQ locaux»](#), à la page 144

Vous pouvez administrer des objets IBM MQ locaux pour prendre en charge les programmes d'application qui utilisent l'interface MQI (Message Queue Interface).

[«Administration des objets IBM MQ éloignés»](#), à la page 199

Cette section explique comment administrer des objets IBM MQ sur un gestionnaire de files d'attente éloignées à l'aide de commandes MQSC et comment utiliser des objets de file d'attente éloignée pour contrôler la destination des messages et des messages de réponse.

Information associée

[Concepts d'IBM MQ for z/OS](#)

[Planification](#)

[Planification de votre environnement IBM MQ sur z/OS](#)

[Configuration](#)

[Configuration en cours z/OS](#)

[Références relatives aux formats de commandes programmables](#)

[Référence MQSC](#)

[Utilisation des utilitaires IBM MQ for z/OS](#)

z/OS

Emission de commandes sur IBM MQ for z/OS

Vous pouvez utiliser des commandes de script IBM MQ (MQSC) en mode de traitement par lots ou en mode interactif pour contrôler un gestionnaire de files d'attente.

IBM MQ for z/OS prend en charge les commandes MQSC, qui peuvent être émises à partir des sources suivantes:

- Console z/OS ou équivalent (tel que SDSF/TSO).
- Fichiers en entrée d'initialisation.
- L'utilitaire de traitement par lots fourni, CSQUTIL, traite une liste de commandes dans un fichier séquentiel.
- Une application disposant des droits appropriés, en envoyant une commande en tant que message à la file d'attente d'entrée de commande. L'application peut être l'une des suivantes:
 - Un programme de région de traitement par lots
 - Une application CICS
 - Une application IMS
 - Une application TSO
 - Un programme d'application ou un utilitaire sur un autre système IBM MQ

Le [Tableau 24](#), à la page 339 récapitule les commandes MQSC et les sources à partir desquelles elles peuvent être émises.

La plupart des fonctionnalités de ces commandes sont disponibles de manière pratique dans les panneaux d'opérations et de commandes IBM MQ for z/OS .

Les modifications apportées aux définitions de ressources d'un gestionnaire de files d'attente à l'aide des commandes (directement ou indirectement) sont conservées lors des redémarrages du sous-système IBM MQ .

IBM MQ for z/OS prend également en charge les commandes PCF (Programmable Command Format). Elles simplifient la création d'applications pour l'administration de IBM MQ. Les commandes MQSC sont sous forme de texte lisible par l'utilisateur, tandis que PCF permet aux applications de créer des demandes et de lire les réponses sans avoir à analyser les chaînes de texte. Comme les commandes MQSC, les applications émettent des commandes PCF en les envoyant sous forme de messages à la file d'attente d'entrée de commandes. Pour plus d'informations sur l'utilisation des commandes PCF et pour plus de détails sur ces commandes, voir la documentation [Informations de référence sur les formats de commande programmables](#) .

Définitions privées et globales sous IBM MQ for z/OS

Lorsque vous définissez un objet sur IBM MQ for z/OS, vous pouvez choisir si vous souhaitez partager cette définition avec d'autres gestionnaires de files d'attente (une définition *globale*) ou si la définition d'objet doit être utilisée par un seul gestionnaire de files d'attente (une définition *privée*). Il s'agit de l'objet *disposition*.

Définition globale

Si votre gestionnaire de files d'attente appartient à un groupe de partage de files d'attente, vous pouvez choisir de partager les définitions d'objet que vous créez avec les autres membres du groupe. Cela signifie que vous devez définir un objet une seule fois, ce qui réduit le nombre total de définitions requises pour l'ensemble du système.

Les définitions d'objet global sont conservées dans un *référentiel partagé* (base de données partagée Db2) et sont disponibles pour tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Ces objets ont une disposition GROUP.

Définition privée

Si vous souhaitez créer une définition d'objet requise par un seul gestionnaire de files d'attente ou si votre gestionnaire de files d'attente n'est pas membre d'un groupe de partage de files d'attente, vous pouvez créer des définitions d'objet qui ne sont pas partagées avec d'autres membres d'un groupe de partage de files d'attente.

Les définitions d'objet privé sont conservées dans l'ensemble de pages zéro du gestionnaire de files d'attente de définition. Ces objets ont une disposition de QMGR.

Vous pouvez créer des définitions privées pour tous les types d'objets IBM MQ , à l'exception des structures d'unité de couplage (c'est-à-dire, les canaux, les listes de noms, les définitions de processus, les files d'attente, les gestionnaires de files d'attente, les définitions de classe de stockage et les objets d'informations d'authentification) et des définitions globales pour tous les types d'objet, à l'exception des gestionnaires de files d'attente.

IBM MQ copie automatiquement la définition d'un objet de groupe dans l'ensemble de pages zéro de chaque gestionnaire de files d'attente qui l'utilise. Vous pouvez modifier temporairement la copie de la définition si vous le souhaitez, et IBM MQ vous permet d'actualiser les copies de l'ensemble de pages à partir de la copie du référentiel si nécessaire.

IBM MQ tente toujours d'actualiser les copies de l'ensemble de pages à partir de la copie du référentiel au démarrage (pour les commandes de canal, cette opération est effectuée lorsque l'initiateur de canal redémarre) ou si l'objet de groupe est modifié.

Remarque : La copie de la définition est actualisée à partir de la définition du groupe, uniquement si la définition du groupe a été modifiée après la création de la copie de la définition.

Cela garantit que les copies de l'ensemble de pages reflètent la version dans le référentiel, y compris les modifications apportées lorsque le gestionnaire de files d'attente était inactif. Les copies

sont actualisées en générant des commandes DEFINE REPLACE. Par conséquent, dans certaines circonstances, l'actualisation n'est pas effectuée, par exemple:

- Si une copie de la file d'attente est ouverte, une actualisation qui modifie l'utilisation de la file d'attente échoue.
- Si une copie d'une file d'attente contient des messages, une actualisation qui supprime cette file d'attente échoue.
- Si une copie d'une file d'attente nécessite ALTER avec FORCE pour la modifier.

Dans ces circonstances, l'actualisation n'est pas effectuée sur cette copie, mais sur les copies de tous les autres gestionnaires de files d'attente.

Si le gestionnaire de files d'attente est arrêté, puis redémarré en autonome, toutes les copies locales des objets sont supprimées, sauf si, par exemple, des messages sont associés à la file d'attente.

Il existe une troisième disposition d'objet qui s'applique uniquement aux files d'attente locales. Vous pouvez ainsi créer des files d'attente partagées. La définition d'une file d'attente partagée est conservée dans le référentiel partagé et est disponible pour tous les gestionnaires de files d'attente du groupe de partage de files d'attente. En outre, les messages d'une file d'attente partagée sont également disponibles pour tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Cette procédure est décrite dans [Files d'attente partagées et groupes de partage de files d'attente](#). Les files d'attente partagées ont une disposition d'objet SHARED.

Le tableau suivant récapitule l'effet des options de disposition d'objet pour les gestionnaires de files d'attente démarrés en autonome et en tant que membre d'un groupe de partage de files d'attente.

| Disposition | Gestionnaire de files d'attente autonome | Membre d'un groupe de partage de files d'attente |
|-------------|--|---|
| QMGR | Définition d'objet contenue dans l'ensemble de pages zéro. | Définition d'objet contenue dans l'ensemble de pages zéro. |
| GROUPE | Non autorisé | Définition d'objet contenue dans le référentiel partagé. Copie locale conservée sur l'ensemble de pages zéro de chaque gestionnaire de files d'attente du groupe. |
| PARTAGE | Non autorisé | Définition de file d'attente contenue dans le référentiel partagé. Messages disponibles pour tous les gestionnaires de files d'attente du groupe. |

Manipulation des définitions globales

Si vous souhaitez modifier la définition d'un objet contenu dans le référentiel partagé, vous devez indiquer si vous souhaitez modifier la version dans le référentiel ou la copie locale dans le jeu de pages zéro. Pour ce faire, utilisez la disposition d'objet dans le cadre de la commande.

Transmission de commandes à différents gestionnaires de files d'attente sous z/OS

Vous pouvez utiliser la *portée de la commande* pour contrôler le gestionnaire de files d'attente sur lequel la commande s'exécute.

Vous pouvez choisir d'exécuter une commande sur le gestionnaire de files d'attente dans lequel elle est entrée ou sur un autre gestionnaire de files d'attente du groupe de partage de files d'attente. Vous pouvez également choisir d'émettre une commande particulière en parallèle sur tous les gestionnaires de files d'attente d'un groupe de partage de files d'attente. Cela est possible pour les commandes MQSC et PCF.

Ceci est déterminé par la *portée de la commande*. La portée de la commande est utilisée avec la disposition d'objet pour déterminer la version d'un objet que vous souhaitez utiliser.

Par exemple, vous pouvez modifier certains attributs d'un objet dont la définition est conservée dans le référentiel partagé.

- Vous pouvez modifier la version sur un seul gestionnaire de files d'attente et ne pas modifier la version sur le référentiel ou celles utilisées par d'autres gestionnaires de files d'attente.
- Vous pouvez modifier la version dans le référentiel partagé pour les utilisateurs futurs, mais laisser les copies existantes inchangées.
- Vous souhaitez peut-être modifier la version dans le référentiel partagé, mais vous souhaitez également que vos modifications soient répercutées immédiatement sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente qui contiennent une copie de l'objet sur leur ensemble de pages zéro.

Utilisez la portée de la commande pour indiquer si la commande est exécutée sur ce gestionnaire de files d'attente, sur un autre gestionnaire de files d'attente ou sur tous les gestionnaires de files d'attente. Utilisez la disposition d'objet pour indiquer si l'objet que vous manipulez se trouve dans le référentiel partagé (objet de groupe) ou s'il s'agit d'une copie locale sur le jeu de pages zéro (objet de gestionnaire de files d'attente).

Il n'est pas nécessaire de spécifier la portée de la commande et la disposition de l'objet pour utiliser une file d'attente partagée car chaque gestionnaire de files d'attente du groupe de partage de files d'attente traite la file d'attente partagée comme une file d'attente unique.

Récapitulatif des commandes pour IBM MQ for z/OS

Utilisez cette rubrique comme référence des commandes MQSC et PCF principales.

Le [Tableau 23](#), à la page 338 récapitule les commandes MQSC et PCF disponibles sur IBM MQ for z/OS pour modifier, définir, supprimer et afficher des objets IBM MQ.

| Commande MQSC | ALTER | DEFINIR | affichage | SUPPRIMER |
|---------------|---------|----------------|---------------|-----------|
| Commande PCF | Changer | Créer / Copier | Interrogation | Supprimer |
| AUTHINFO | X | X | X | X |
| STATUT FC | | | X | |
| CFSTRUCT | X | X | X | X |
| Canal | X | X | X | X |
| STATUT | | | X | |
| NAMELIST | X | X | X | X |
| PROCESS | X | X | X | X |
| QALIAS | M | M | M | M |
| QCLUSTER | | | M | |
| QLOCAL | M | M | M | M |
| QMGR | X | | X | |
| QMODEL | M | M | M | M |
| QREMOTE | M | M | M | M |
| QUEUE | P | P | X | P |

Tableau 23. Récapitulatif des principales commandes MQSC et PCF par type d'objet (suite)

| Commande MQSC | ALTER | DEFINIR | affichage | SUPPRIMER |
|---------------|-------|---------|-----------|-----------|
| QSTATUS | | | X | |
| STGCLASS | X | X | X | X |

Symboles de la clé de la table:

- M = MQSC uniquement
- P = PCF uniquement
- X = les deux

Il existe de nombreuses autres commandes MQSC et PCF qui vous permettent de gérer d'autres ressources IBM MQ et d'effectuer d'autres actions en plus de celles décrites dans [Tableau 23](#), à la page 338.

La [Tableau 24](#), à la page 339 montre chaque commande MQSC et où chaque commande peut être émise à partir de:

- Fichier en entrée d'initialisation CSQINP1
- Fichier d'entrée d'initialisation CSQINP2
- Console z/OS (ou équivalent)
- SYSTEME SYSTEM.COMMAND.INPUT (à partir d'applications, de CSQUTIL ou du fichier d'entrée d'initialisation CSQINPX)

Tableau 24. Sources à partir desquelles les commandes MQSC doivent être exécutées

| Commande | CSQINP1 | CSQINP2 | console z/OS | File d'attente d'entrée de commandes et serveur |
|-----------------------|---------|---------|--------------|---|
| ALTER AUTHINFO | | X | X | X |
| ALTER BUFFPOOL | | X | X | X |
| ALTER CFSTRUCT | | X | X | X |
| ALTER CHANNEL | | X | X | X |
| ALTER NAMELIST | | X | X | X |
| ALTER PSID | | | X | X |
| MODIFIER LE PROCESSUS | | X | X | X |
| ALTER QALIAS | | X | X | X |
| ALTER QLOCAL | | X | X | X |
| ALTER QMGR | | X | X | X |
| ALTER QMODEL | | X | X | X |
| MODIFIER QREMOTE | | X | X | X |
| MODIFIER SECURITE | X | X | X | X |
| ALTER STGCLASS | | X | X | X |
| ALTER SUB | | X | X | X |

Tableau 24. Sources à partir desquelles les commandes MQSC doivent être exécutées (suite)

| Commande | CSQINP1 | CSQINP2 | console z/OS | File d'attente d'entrée de commandes et serveur |
|-----------------------------------|---------|---------|--------------|---|
| ALTER TOPIC | | X | X | X |
| ALTER TRACE | X | X | X | X |
| ARCHIVE LOG | X | X | X | X |
| STRUCTURE CFSTRUCT DE SAUVEGARDE | | | X | X |
| QLOCAL CLEAR | | X | X | X |
| DEFINE AUTHINFO | | X | X | X |
| DEFINIR LE POOL DE MÉMOIRE TAMPON | X | X | | |
| DEFINE CFSTRUCT | | X | X | X |
| De la définition d'un canal | | X | X | X |
| DEFINE LOG | | | X | X |
| DEFINE NAMELIST | | X | X | X |
| DEFINE PROCESSUS | | X | X | X |
| DEFINE PSID | X | | X | X |
| DEFINE QALIAS | | X | X | X |
| DEFINE QLOCAL | | X | X | X |
| DEFINIR QMODEL | | X | X | X |
| DEFINIR QREMOTE | | X | X | X |
| DEFINE STGCLASS | | X | X | X |
| DEFINE SUB | | | X | X |
| DEFINE TOPIC | | X | X | X |
| DELETE AUTHINFO | | X | X | X |
| DELETE BUFFPOOL | | | X | X |
| DELETE CFSTRUCT | | X | X | X |
| Supprimer le canal | | | X | X |
| Supprimer une liste de noms | | X | X | X |
| Supprimer un processus | | X | X | X |
| PSID DELETE | | | X | X |
| DELETE QALIAS | | X | X | X |
| SUPPRIMER QLOCAL | | X | X | X |
| SUPPRIMER QMODEL | | X | X | X |
| SUPPRIMER QREMOTE | | X | X | X |

Tableau 24. Sources à partir desquelles les commandes MQSC doivent être exécutées (suite)

| Commande | CSQINP1 | CSQINP2 | console z/OS | File d'attente d'entrée de commandes et serveur |
|---|---------|---------|--------------|---|
| DELETE STGCLASS | | X | X | X |
| SUPPRIMER DES SOUS | | X | X | X |
| Supprimer la rubrique | | X | X | X |
| AFFICHER L'ARCHIVE | X | X | X | X |
| INFORMATIONS D'AUTHENTIFICATION D'AFFICHAGE | | X | X | X |
| STATUT CF AFFICHAGE | | | X | X |
| DISPLAY CFSTRUCT | | X | X | X |
| CANAL D'AFFICHAGE | | X | X | X |
| DISPLAY CHSTATUS | | | X | X |
| DISPLAY CLUSQMGR | | | X | X |
| DISPLAY CMDSERV | X | X | X | X |
| DISPLAY CONN | | X | X | X |
| AFFICHAGE DE CHINIT | | X | X | X |
| Afficher le groupe | | X | X | X |
| JOURNAL D'AFFICHAGE | X | X | X | X |
| AFFICHAGE DE LA LISTE DE NOMS | | X | X | X |
| PROCESSUS D'AFFICHAGE | | X | X | X |
| AFFICHER QALIAS | | X | X | X |
| AFFICHER QCLUSTER | | X | X | X |
| DISPLAY QLOCAL | | X | X | X |
| DISPLAY QMGR | | X | X | X |
| AFFICHEZ QMODEL | | X | X | X |
| AFFICHER QREMOTE | | X | X | X |
| STATUT DE LA FILE D'ATTENTE D'AFFICHAGE | | X | X | X |
| DISPLAY QUEUE | | X | X | X |
| SECURITE D'AFFICHAGE | | | X | X |
| DISPLAY STGCLASS | | X | X | X |
| DISPLAY SUB | | X | X | X |
| DISPLAY TOPIC | | X | X | X |
| DISPLAY SYSTEM | X | X | X | X |

Tableau 24. Sources à partir desquelles les commandes MQSC doivent être exécutées (suite)

| Commande | CSQINP1 | CSQINP2 | console z/OS | File d'attente d'entrée de commandes et serveur |
|---|---------|---------|--------------|---|
| afficher l'unité d'exécution | | X | X | X |
| AFFICHER TRACE | X | X | X | X |
| Affichage d'informations | | X | X | X |
| MOVE QLOCAL | | X | X | X |
| Envoyer une commande PING à un canal | | | X | X |
| Récupérer un fichier d'amorce | X | X | X | X |
| RECUPERER CFSTRUCT | | | X | X |
| Actualiser le cluster | | X | X | X |
| ACTUALISEZ LE GESTIONNAIRE DE FILES D'ATTENTE | | X | X | X |
| REFRESH SECURITY | | X | X | X |
| Réinitialisation du canal | | | X | X |
| Réinitialisation d'un cluster | | X | X | X |
| RESET QSTATS | | X | X | X |
| REINITIALISER TPIPE | | | X | X |
| Résolution du canal | | | X | X |
| Résoudre le statut en attente de validation | | X | X | X |
| RESUME QMGR | | | X | X |
| SECURITE RVERIFY | | X | X | X |
| Définir une archive | X | X | X | X |
| Définir un journal | X | X | X | X |
| Définir un système | X | X | X | X |
| Démarrer un canal | | | X | X |
| START CHINIT | | X | X | X |
| DEMARRAGE de CMDSERV | X | X | X | |
| Démarrer le programme d'écoute | | | X | X |
| START QMGR | | | X | |
| Démarrer une trace | X | X | X | X |
| Arrêter le canal | | | X | X |
| STOP CHINIT | | | X | X |
| ARRETER CMDSERV | X | X | X | |

Tableau 24. Sources à partir desquelles les commandes MQSC doivent être exécutées (suite)

| Commande | CSQINP1 | CSQINP2 | console z/OS | File d'attente d'entrée de commandes et serveur |
|-------------------------------|---------|---------|--------------|---|
| Arrêter le programme d'écoute | | | X | X |
| STOP QMGR | | | X | X |
| Arrêter le traçage | X | X | X | X |
| SUSPEND QMGR | | | X | X |

Dans les commandes MQSC, chaque description de commande identifie les sources à partir desquelles cette commande peut être exécutée.

Commandes d'initialisation pour IBM MQ for z/OS

Les commandes d'initialisation peuvent être utilisées pour contrôler le démarrage du gestionnaire de files d'attente.

Les commandes des fichiers d'entrée d'initialisation sont traitées lorsque IBM MQ est initialisé au démarrage du gestionnaire de files d'attente. Trois types de commande peuvent être émis à partir des fichiers d'entrée d'initialisation:

- Commandes permettant de définir des entités IBM MQ qui ne peuvent pas être définies ailleurs, par exemple DEFINE BUFFPOOL.

Ces commandes doivent résider dans le fichier identifié par le nom de définition de données CSQINP1. Ils sont traités avant la phase de redémarrage de l'initialisation. Ils ne peuvent pas être émis via la console, les panneaux d'opérations et de contrôle ou un programme d'application. Les réponses à ces commandes sont écrites dans le fichier séquentiel auquel vous faites référence dans l'instruction CSQOUT1 de la procédure de tâche démarrée.

- Commandes permettant de définir les objets IBM MQ qui sont récupérables après le redémarrage. Ces définitions doivent être spécifiées dans le fichier identifié par le nom de définition de données CSQINP2. Ils sont stockés dans l'ensemble de pages zéro. CSQINP2 est traité après la phase de redémarrage de l'initialisation. Les réponses à ces commandes sont écrites dans le fichier séquentiel auquel vous faites référence dans l'instruction CSQOUT2 de la procédure de tâche démarrée.
- Commandes permettant de manipuler des objets IBM MQ. Ces commandes doivent également être spécifiées dans le fichier identifié par le nom symbolique CSQINP2. Par exemple, l'exemple fourni par IBM MQ contient une commande ALTER QMGR pour spécifier une file d'attente de rebut pour le sous-système. La réponse à ces commandes est écrite dans le fichier de sortie CSQOUT2.

Remarque : Si des objets IBM MQ sont définis dans CSQINP2, IBM MQ tente de les redéfinir chaque fois que le gestionnaire de files d'attente est démarré. Si les objets existent déjà, la tentative de les définir échoue. Si vous devez définir vos objets dans CSQINP2, vous pouvez éviter ce problème en utilisant le paramètre REPLACE des commandes DEFINE. Toutefois, ce paramètre remplace les modifications apportées lors de l'exécution précédente du gestionnaire de files d'attente.

Des exemples de membres de fichier d'initialisation sont fournis avec IBM MQ for z/OS. Ils sont décrits dans [Exemples de définitions fournis avec IBM MQ](#).

Commandes d'initialisation pour la mise en file d'attente répartie

Vous pouvez également utiliser le fichier d'initialisation CSQINP2 pour la commande START CHINIT. Si vous avez besoin d'une série d'autres commandes pour définir votre environnement de mise en file d'attente répartie (par exemple, démarrage des programmes d'écoute), IBM MQ fournit un troisième

fichier d'entrée d'initialisation, appelé CSQINPX, qui est traité dans le cadre de la procédure de tâche démarrée de l'initiateur de canal.

Les commandes MQSC contenues dans le fichier sont exécutées à la fin de l'initialisation de l'initiateur de canal et la sortie est écrite dans le fichier spécifié par l'instruction de définition de données CSQOUTX. Vous pouvez utiliser le fichier d'initialisation CSQINPX pour démarrer les programmes d'écoute, par exemple.

Un exemple de membre de fichier d'initialisation d'initiateur de canal est fourni avec IBM MQ for z/OS. Il est décrit dans [Exemples de définitions fournis avec IBM MQ](#).

Commandes d'initialisation pour la publication / abonnement

Si vous avez besoin d'une série de commandes pour définir votre environnement de publication / abonnement (par exemple, lors de la définition d'abonnements), IBM MQ fournit un quatrième fichier d'entrée d'initialisation, appelé CSQINPT.

Les commandes MQSC contenues dans le fichier sont exécutées à la fin de l'initialisation de la fonction de publication / abonnement et la sortie est écrite dans le fichier spécifié par l'instruction de définition de données CSQOUTT. Vous pouvez utiliser l'ensemble de données d'initialisation CSQINPT pour définir des abonnements, par exemple.

Un exemple de membre de fichier d'initialisation de publication / abonnement est fourni avec IBM MQ for z/OS. Il est décrit dans [Exemples de définitions fournis avec IBM MQ](#).

z/OS

Les utilitaires IBM MQ for z/OS

IBM MQ for z/OS fournit un ensemble de programmes utilitaires que vous pouvez utiliser pour faciliter l'administration du système.

IBM MQ for z/OS fournit un ensemble de programmes utilitaires pour vous aider à effectuer diverses tâches d'administration, notamment:

- Gérez les règles de sécurité des messages.
- Effectuer des tâches de sauvegarde, de restauration et de réorganisation.
- Exécutez des commandes et des définitions d'objet de processus.
- Générez des exits de conversion de données.
- Modifiez le fichier d'amorçage.
- Répertoire les informations sur les journaux.
- Imprimez les journaux.
- Configurez des tables Db2 et d'autres utilitaires Db2 .
- Traiter les messages de la file d'attente des messages non livrés.

Utilitaire de stratégie de sécurité des messages

L'utilitaire de règles de sécurité des messages (CSQOUTIL) s'exécute en tant qu'utilitaire autonome pour gérer les règles de sécurité des messages. Pour plus d'informations, voir [Utilitaire de stratégie de sécurité des messages \(CSQOUTIL\)](#) .

L'utilitaire CSQUTIL

Il s'agit d'un programme utilitaire fourni pour vous aider à effectuer des tâches de sauvegarde, de restauration et de réorganisation. Pour plus d'informations, voir [L'utilitaire CSQUTIL](#) .

L'utilitaire d'exit de conversion de données

L'utilitaire d'exit de conversion de données IBM MQ for z/OS (**CSQUCVX**) s'exécute en tant qu'utilitaire autonome pour créer des routines d'exit de conversion de données.

Utilitaire de modification de l'inventaire des journaux

Programme utilitaire d'inventaire des journaux de modifications IBM MQ for z/OS (**CSQJU003**) s'exécute en tant qu'utilitaire autonome pour modifier le fichier d'amorce. Vous pouvez utiliser l'utilitaire pour exécuter les fonctions suivantes:

- Ajouter ou supprimer des fichiers journaux actifs ou d'archivage.
- Indiquez des mots de passe pour les journaux d'archivage.

Utilitaire d'impression de mappe de journaux

Le programme utilitaire d'impression de mappe de journal IBM MQ for z/OS (**CSQJU004**) s'exécute en tant qu'utilitaire autonome pour répertorier les informations suivantes:

- Nom de fichier journal et association d'adresse relative en octets de journal pour les copies de tous les fichiers journaux actifs et d'archivage. Si la double consignation n'est pas active, il n'y a qu'une seule copie des fichiers.
- Fichiers journaux actifs disponibles pour les nouvelles données de journal.
- Contenu de la file d'attente des enregistrements de point de contrôle dans le fichier d'amorce.
- Contenu de l'enregistrement de l'historique des commandes du journal d'archivage.
- Horodatages du système et de l'utilitaire.

Utilitaire d'impression de journal

L'utilitaire d'impression de journal (**CSQ1LOGP**) est exécuté en tant qu'utilitaire autonome. Vous pouvez exécuter l'utilitaire en spécifiant:

- Un fichier d'amorçage (BSDS)
- Journaux actifs (sans fichier d'amorce)
- Journaux d'archivage (sans fichier d'amorce)

Utilitaire de groupe de partage de files d'attente

Programme utilitaire de groupe de partage de files d'attente (**CSQ5PQSG**) s'exécute en tant qu'utilitaire autonome pour configurer des tables Db2 et effectuer d'autres tâches Db2 requises pour les groupes de partage de files d'attente.

Utilitaire de préformatage des journaux actifs

L'utilitaire de préformatage des journaux actifs (**CSQJUFMT**) formate les fichiers journaux actifs avant qu'ils ne soient utilisés par un gestionnaire de files d'attente. Si les fichiers journaux actifs sont préformatés par l'utilitaire, les performances d'écriture de journal sont améliorées lors du premier passage du gestionnaire de files d'attente dans les journaux actifs.

Utilitaire de gestionnaire de files d'attente de rebut

Programme utilitaire de gestionnaire de files d'attente de rebut (**CSQUDLQH**) s'exécute en tant qu'utilitaire autonome. Il vérifie les messages qui se trouvent dans la file d'attente de rebut et les traite conformément à un ensemble de règles que vous fournissez à l'utilitaire.

Utilitaire CSQUTIL pour IBM MQ for z/OS

Le programme utilitaire CSQUTIL est fourni avec IBM MQ for z/OS pour vous aider à effectuer des tâches de sauvegarde, de restauration et de réorganisation, et à émettre des commandes et des définitions d'objet de processus.

Pour plus d'informations sur le programme utilitaire CSQUTIL, voir [Programme utilitaire IBM MQ \(CSQUTIL\)](#). A l'aide de ce programme utilitaire, vous pouvez appeler les fonctions suivantes:

command

Permet d'émettre des commandes MQSC, d'enregistrer des définitions d'objet et de créer des fichiers de définition de canal client.

COPIER

Pour lire le contenu d'une file d'attente de messages IBM MQ for z/OS nommée ou le contenu de toutes les files d'attente d'un ensemble de pages nommé, placez-les dans un fichier séquentiel et conservez la file d'attente d'origine.

PAGE DE COPIE

Permet de copier des ensembles de pages entiers dans des ensembles de pages plus volumineux.

EMPTY

Pour supprimer le contenu d'une file d'attente de messages IBM MQ for z/OS nommée ou le contenu de toutes les files d'attente d'un ensemble de pages nommé, conservez les définitions des files d'attente.

Format

Pour formater les ensembles de pages IBM MQ for z/OS .

CHARGEMENT

Pour restaurer le contenu d'une file d'attente de messages IBM MQ for z/OS nommée ou le contenu de toutes les files d'attente d'un ensemble de pages nommé à partir d'un fichier séquentiel créé par la fonction COPY.

PageInfo

Permet d'extraire des informations d'ensemble de pages à partir d'un ou de plusieurs ensembles de pages.

RESETPAGE

Pour copier des ensembles de pages entiers dans d'autres ensembles de données d'ensemble de pages et réinitialiser les informations de journal dans la copie.

SCOPY

Permet de copier le contenu d'une file d'attente dans un fichier alors que le gestionnaire de files d'attente est hors ligne.

SDEFS

Permet de générer un ensemble de commandes de définition pour les objets lorsque le gestionnaire de files d'attente est hors ligne.

CHARGEMENT

Pour restaurer des messages à partir du fichier de destination d'une opération COPY ou SCOPY antérieure. SLOAD traite une file d'attente unique.

Bascule

Permet de basculer ou d'interroger la file d'attente de transmission associée aux canaux émetteurs de cluster.

XPARM

Pour convertir un module de chargement de paramètre d'initiateur de canal en attributs de gestionnaire de files d'attente (à des fins de migration).

fonctionnement IBM MQ for z/OS

Utilisez ces procédures de base pour utiliser IBM MQ for z/OS.

Vous pouvez également effectuer les opérations décrites dans cette section à l'aide de IBM MQ Explorer, qui est distribué avec IBM MQ for Windows, IBM MQ for Linux (plateformes x86 et x86-64) et SupportPac MS0T. Pour plus d'informations, voir [«Administration à l'aide de IBM MQ Explorer»](#), à la page 135 et [IBM Support & downloads](#).

Cette section contient des informations sur les rubriques suivantes:

Emission de commandes de gestionnaire de files d'attente sous z/OS

Vous pouvez exécuter des commandes de contrôle IBM MQ à partir d'une console z/OS ou à l'aide du programme utilitaire CSQUTIL. Les commandes peuvent utiliser la chaîne de préfixe de commande (CPF) pour indiquer le sous-système IBM MQ qui traite la commande.

Vous pouvez contrôler la plupart de l'environnement opérationnel de IBM MQ à l'aide des commandes IBM MQ . IBM MQ for z/OS prend en charge les types MQSC et PCF de ces commandes. Cette rubrique explique comment spécifier des attributs à l'aide de commandes MQSC. Elle fait donc référence à ces commandes et attributs à l'aide de leurs noms de commande MQSC et non de leurs noms PCF. Pour plus de détails sur la syntaxe des commandes MQSC, voir [Les commandes MQSC](#). Pour plus de détails sur la syntaxe des commandes PCF, voir [«Utilisation des formats de commande programmables IBM MQ»](#), à la page 23. Si vous êtes un utilisateur dûment autorisé, vous pouvez exécuter les commandes IBM MQ à partir de:

- Fichiers en entrée d'initialisation (décrits dans [«Commandes d'initialisation pour IBM MQ for z/OS»](#), à la page 343).
- Une console z/OS , ou équivalent, telle que SDSF
- La routine de commande get maître z/OS , MGCRE (SVC 34)
- L'utilitaire IBM MQ , CSQUTIL (décrit dans le programme utilitaire [IBM MQ](#).)
- Une application utilisateur, qui peut être:
 - Un programme CICS
 - Un programme TSO
 - Un programme batch z/OS
 - Un programme IMS

Pour plus d'informations à ce sujet, voir [«Ecriture de programmes pour administrer IBM MQ for z/OS»](#), à la page 368 .

La plupart des fonctionnalités de ces commandes sont fournies de manière pratique par les panneaux d'opérations et de contrôle, accessibles depuis TSO et ISPF, et décrits dans [«Panneaux d'opérations et de contrôle pour IBM MQ for z/OS»](#), à la page 353.

Pour plus d'informations, voir

- [«Emission de commandes à partir d'une console z/OS ou de son équivalent»](#), à la page 347
 - [Chaînes de préfixe de commande](#)
 - [Utilisation de la console z/OS pour émettre des commandes](#)
 - [Réponses de la commande](#)
- [Génération de commandes à partir du programme utilitaire CSQUTIL](#)

Emission de commandes à partir d'une console z/OS ou de son équivalent

Vous pouvez exécuter toutes les commandes IBM MQ à partir d'une console z/OS ou de son équivalent. Vous pouvez également émettre des commandes IBM MQ à partir de n'importe quel endroit où vous

pouvez émettre des commandes z/OS , telles que SDSF , ou à l'aide d'un programme utilisant la macro MGCRC.

La quantité maximale de données pouvant être affichées suite à une commande saisie sur la console est de 32 Ko.

Remarque :

1. Vous ne pouvez pas émettre de commandes IBM MQ à l'aide du format de commande IMS/SSR à partir d'un terminal IMS . Cette fonction n'est pas prise en charge par l'adaptateur IMS .
2. La zone d'entrée fournie par SDSF peut ne pas être suffisamment longue pour certaines commandes, en particulier celles des canaux.

Chaînes de préfixe de commande

Chaque commande IBM MQ doit être précédée d'une chaîne de préfixe de commande (CPF), comme illustré dans la [Figure 41](#), à la [page 348](#).

Etant donné que plusieurs sous-systèmes IBM MQ peuvent s'exécuter sous z/OS, le CPF est utilisé pour indiquer le sous-système IBM MQ qui traite la commande. Par exemple, pour démarrer le gestionnaire de files d'attente pour un sous-système appelé CSQ1, où CPF est ' +CSQ1 ', vous émettez la commande +CSQ1 START QMGR à partir de la console opérateur. Ce CPF doit être défini dans la table des noms de sous-système (pour le sous-système CSQ1). Ceci est décrit dans la rubrique [Définition de chaînes de préfixe de commande \(CPF\)](#). Dans les exemples, la chaîne ' +CSQ1 ' est utilisé comme préfixe de commande.

Utilisation de la console z/OS pour émettre des commandes

Vous pouvez entrer des commandes simples à partir de la console z/OS , par exemple, la commande DISPLAY dans [Figure 41](#), à la [page 348](#). Toutefois, pour les commandes complexes ou pour les ensembles de commandes que vous émettez fréquemment, les autres méthodes d'émission de commandes sont plus efficaces.

```
+CSQ1 DISPLAY QUEUE(TRANSMIT.QUEUE.PROD) TYPE(QLLOCAL)
```

Figure 41. Emission d'une commande DISPLAY à partir de la console z/OS

Réponses de commande

Les réponses directes aux commandes sont envoyées à la console qui a émis la commande. IBM MQ prend en charge la fonction *Extended Console Support* (EMCS) disponible dans z/OS. Par conséquent, les consoles avec des ID de 4 octets peuvent être utilisées. De plus, toutes les commandes à l'exception de START QMGR et STOP QMGR prennent en charge l'utilisation de jetons de commande et de réponse (CARTs) lorsque la commande est émise par un programme utilisant la macro MGCRC.

Emission de commandes à partir du programme utilitaire CSQUTIL

Vous pouvez émettre des commandes à partir d'un fichier séquentiel à l'aide de la fonction COMMAND du programme utilitaire CSQUTIL. Cet utilitaire transfère les commandes, en tant que messages, dans la *file d'attente d'entrée des commandes système* et attend la réponse, qui est imprimée avec les commandes d'origine dans SYSPRINT. Pour plus de détails, voir [Programme utilitaire IBM MQ](#).

Démarrage et arrêt d'un gestionnaire de files d'attente sous z/OS

Cette rubrique présente l'arrêt et le démarrage d'un gestionnaire de files d'attente.

Cette section explique comment démarrer et arrêter un gestionnaire de files d'attente. Il contient des informations sur les rubriques suivantes:

- «Avant de commencer IBM MQ», à la page 349
- «Démarrage d'un gestionnaire de files d'attente», à la page 349
- «Arrêt d'un gestionnaire de files d'attente», à la page 351

Le démarrage et l'arrêt d'un gestionnaire de files d'attente sont relativement simples. Lorsqu'un gestionnaire de files d'attente s'arrête dans des conditions normales, sa dernière action consiste à prendre un point de contrôle d'arrêt. Ce point de contrôle et les journaux fournissent au gestionnaire de files d'attente les informations dont il a besoin pour redémarrer.

Cette section contient des informations sur les commandes START et STOP, ainsi qu'une brève présentation du démarrage après un arrêt anormal.

Avant de commencer IBM MQ

Une fois que vous avez installé IBM MQ, il est défini en tant que sous-système z/OS formel. Ce message s'affiche lors d'un IPL (procédure de chargement initial) de z/OS:

```
CSQ3110I +CSQ1 CSQ3UR00 - SUBSYSTEM ssnm INITIALIZATION COMPLETE
```

où *ssnm* est le nom du sous-système IBM MQ .

A partir de maintenant, vous pouvez démarrer le gestionnaire de files d'attente pour ce sous-système à partir de n'importe quelle console z/OS qui a été autorisée à émettre des commandes de contrôle système ; c'est-à-dire un groupe de commandes SYS z/OS . Vous devez émettre la commande START à partir de la console autorisée, vous ne pouvez pas l'émettre via JES ou TSO.

Si vous utilisez des groupes de partage de files d'attente, vous devez d'abord démarrer RRS, puis Db2, avant de démarrer le gestionnaire de files d'attente.

Démarrage d'un gestionnaire de files d'attente

Vous démarrez un gestionnaire de files d'attente à l'aide de la commande START QMGR. Toutefois, vous ne pouvez pas utiliser la commande START sans disposer des droits appropriés. Pour plus d'informations sur la sécurité IBM MQ , voir [Configuration de la sécurité sur z/OS](#) . Le [Figure 42](#), à la page 349 présente des exemples de la commande START. (N'oubliez pas que vous devez préfixer une commande IBM MQ avec une chaîne de préfixe de commande (CPF).)

```
+CSQ1 START QMGR
+CSQ1 START QMGR PARM(NEWLOG)
```

Figure 42. Démarrage du gestionnaire de files d'attente à partir d'une console z/OS

Pour plus d'informations sur la syntaxe de la commande START QMGR, voir [START QMGR](#) .

Vous ne pouvez pas exécuter le gestionnaire de files d'attente en tant que travail par lots ou le démarrer à l'aide de la commande z/OS START. Ces méthodes sont susceptibles de démarrer un espace adresse pour IBM MQ qui se termine alors de manière anormale. Vous ne pouvez pas non plus démarrer un gestionnaire de files d'attente à partir du programme utilitaire CSQUTIL ou d'une application utilisateur similaire.

Vous pouvez toutefois démarrer un gestionnaire de files d'attente à partir d'un programme doté de droits APF en transmettant une commande START QMGR au service z/OS MGCRC (SVC 34).

Si vous utilisez des groupes de partage de files d'attente, les systèmes Db2 et RRS associés doivent être actifs lorsque vous démarrez le gestionnaire de files d'attente.

Options de démarrage

Lorsque vous démarrez un gestionnaire de files d'attente, un module de paramètres système est chargé. Vous pouvez spécifier le nom du module de paramètres système de l'une des deux manières suivantes:

- Avec le paramètre PARM de la commande /cpf START QMGR, par exemple

```
/cpf START QMGR PARM(CSQ1ZPRM)
```

- Avec un paramètre dans la procédure de démarrage, par exemple, codez l'instruction JCL EXEC comme suit:

```
//MQM EXEC PGM=CSQYASCP,PARM='ZPARM(CSQ1ZPRM)'
```

Un module de paramètres système fournit les informations spécifiées lors de la personnalisation du gestionnaire de files d'attente.

V 9.0.3 Depuis la IBM MQ 9.0.3, vous pouvez utiliser l'option **QMGRPROD** pour spécifier le produit pour lequel l'utilisation du gestionnaire de files d'attente doit être enregistrée, et l'option **AMSPROD** pour spécifier l'équivalent pour AMS si celui-ci est utilisé. Pour plus de détails sur les valeurs admises, voir la commande MQSC [START QMGR](#).

V 9.0.3 Voici un exemple d'instruction JCL EXEC:

```
//MQM EXEC PGM=CSQYASCP,PARM='QMGRPROD(MQ)'
```

Pour plus d'informations sur l'enregistrement de l'utilisation du produit, voir [z/OS MVS Product Management](#).

Vous pouvez également utiliser l'option ENVPARM pour remplacer un ou plusieurs paramètres dans la procédure JCL du gestionnaire de files d'attente.

Par exemple, vous pouvez mettre à jour la procédure de démarrage de votre gestionnaire de files d'attente de sorte que le nom symbolique CSQINP2 soit une variable. Cela signifie que vous pouvez modifier le nom symbolique CSQINP2 sans modifier la procédure de démarrage. Cette fonction est utile pour l'implémentation des modifications, en fournissant des annulations pour les opérateurs et les opérations du gestionnaire de files d'attente.

Supposons que votre procédure de démarrage pour le gestionnaire de files d'attente CSQ1 ressemble à [Figure 43](#), à la page 351.

```

//CSQ1MSTR PROC INP2=NORM
//MQMESA EXEC PGM=CSQYASCP
//STEPLIB DD DISP=SHR,DSN=thlqual.SCSQANLE
//          DD DISP=SHR,DSN=thlqual.SCSQAUTH
//          DD DISP=SHR,DSN=db2qual.SDSNLOAD
//BDS1 DD DISP=SHR,DSN=myqual.BSDS01
//BDS2 DD DISP=SHR,DSN=myqual.BSDS02
//CSQP0000 DD DISP=SHR,DSN=myqual.PSID00
//CSQP0001 DD DISP=SHR,DSN=myqual.PSID01
//CSQP0002 DD DISP=SHR,DSN=myqual.PSID02
//CSQP0003 DD DISP=SHR,DSN=myqual.PSID03
//CSQINP1 DD DISP=SHR,DSN=myqual.CSQINP(CSQ1INP1)
//CSQINP2 DD DISP=SHR,DSN=myqual.CSQINP(CSQ1&INP2.)
//CSQOUT1 DD SYSOUT=*
//CSQOUT2 DD SYSOUT=*

```

Figure 43. Exemple de procédure de démarrage

Si vous démarrez ensuite votre gestionnaire de files d'attente à l'aide de la commande:

```
+CSQ1 START QMGR
```

CSQINP2 utilisé est un membre appelé CSQ1NORM.

Toutefois, supposez que vous mettiez une nouvelle suite de programmes en production de sorte que lors du prochain démarrage du gestionnaire de files d'attente CSQ1, les définitions CSQINP2 soient extraites du membre CSQ1NEW. Pour ce faire, vous devez démarrer le gestionnaire de files d'attente à l'aide de la commande suivante:

```
+CSQ1 START QMGR ENVPARM('INP2=NEW')
```

et CSQ1NEW serait utilisé à la place de CSQ1NORM. Remarque: z/OS limite les spécifications KEYWORD=value pour les paramètres symboliques (comme dans INP2=NEW) à 255 caractères.

Démarrage après un arrêt anormal

IBM MQ détecte automatiquement si le redémarrage suit un arrêt normal ou un arrêt anormal.

Le démarrage d'un gestionnaire de files d'attente après son arrêt anormal est différent de son démarrage après l'émission de la commande STOP QMGR. Après STOP QMGR, le système termine son travail de manière ordonnée et prend un point de contrôle d'arrêt avant de s'arrêter. Lorsque vous redémarrez le gestionnaire de files d'attente, il utilise les informations du point de contrôle du système et du journal de reprise pour déterminer le statut du système lors de l'arrêt.

Toutefois, si le gestionnaire de files d'attente s'arrête de manière anormale, il s'arrête sans pouvoir terminer son travail ou prendre un point de contrôle d'arrêt. Lorsque vous redémarrez un gestionnaire de files d'attente après une fin anormale, il actualise sa connaissance de son statut lors de l'arrêt à l'aide des informations du journal et vous informe du statut des différentes tâches. Normalement, le processus de redémarrage résout tous les états incohérents. Mais, dans certains cas, vous devez prendre des mesures spécifiques pour résoudre les incohérences.

Messages utilisateur au démarrage

Lorsque vous démarrez correctement un gestionnaire de files d'attente, ce dernier génère un ensemble de messages de démarrage.

Arrêt d'un gestionnaire de files d'attente

Avant d'arrêter un gestionnaire de files d'attente, tous les messages WTOR (Write-to-Operator-with-reply) liés à IBM MQ doivent recevoir des réponses, par exemple, pour obtenir des demandes de journal. Chaque commande dans [Figure 44](#), à la [page 352](#) arrête un gestionnaire de files d'attente en cours d'exécution.

```
+CSQ1  STOP QMGR
+CSQ1  STOP QMGR MODE(QUIESCE)
+CSQ1  STOP QMGR MODE(FORCE)
+CSQ1  STOP QMGR MODE(RESTART)
```

Figure 44. Arrêt d'un gestionnaire de files d'attente

La commande STOP QMGR prend par défaut la valeur STOP QMGR MODE (QUIESCE).

En mode QUIESCE, IBM MQ n'autorise pas la création de nouvelles unités d'exécution de connexion, mais permet la poursuite des unités d'exécution existantes ; il s'arrête uniquement lorsque toutes les unités d'exécution sont terminées. Les applications peuvent demander à être notifiées en cas de mise au repos du gestionnaire de files d'attente. Par conséquent, utilisez le mode QUIESCE dans la mesure du possible pour que les applications qui ont demandé une notification aient la possibilité de se déconnecter. Pour plus d'informations, voir [Que se passe-t-il lors de l'arrêt?](#) .

Si le gestionnaire de files d'attente ne s'arrête pas dans un délai raisonnable en réponse à une commande STOP QMGR MODE (QUIESCE), utilisez la commande DISPLAY CONN pour déterminer s'il existe des unités d'exécution de connexion et effectuez les étapes nécessaires pour arrêter les applications associées. S'il n'y a pas d'unités d'exécution, lancez la commande STOP QMGR MODE (FORCE).

Les commandes STOP QMGR MODE (QUIESCE) et STOP QMGR MODE (FORCE) désenregistrent IBM MQ du gestionnaire de redémarrage automatique MVS (ARM), ce qui empêche ARM de redémarrer automatiquement le gestionnaire de files d'attente. La commande STOP QMGR MODE (RESTART) fonctionne de la même manière que la commande STOP QMGR MODE (FORCE), sauf qu'elle ne désenregistre pas IBM MQ du gestionnaire ARM. Cela signifie que le gestionnaire de files d'attente est éligible pour un redémarrage automatique immédiat.

Si le sous-système IBM MQ n'est pas enregistré auprès d'ARM, la commande STOP QMGR MODE (RESTART) est rejetée et le message suivant est envoyé à la console z/OS :

```
CSQY205I ARM element arm-element is not registered
```

Si ce message n'est pas émis, le gestionnaire de files d'attente est redémarré automatiquement. Pour plus d'informations sur ARM, voir [«Utilisation de z/OS Automatic Restart Manager \(ARM\)»](#), à la [page 434](#).

N'annulez l'espace adresse du gestionnaire de files d'attente que si STOP QMGR MODE (FORCE) n'arrête pas le gestionnaire de files d'attente.

Si un gestionnaire de files d'attente est arrêté en annulant l'espace adresse ou à l'aide de la commande STOP QMGR MODE (FORCE), la cohérence est maintenue avec les systèmes CICS ou IMS connectés. La resynchronisation des ressources est démarrée lorsqu'un gestionnaire de files d'attente redémarre et est terminée lorsque la connexion au système CICS ou IMS est établie.

Remarque : Lorsque vous arrêtez votre gestionnaire de files d'attente, le message IEF352I peut s'afficher. z/OS émet ce message s'il détecte que le fait de ne pas marquer l'espace adresse comme inutilisable entraînerait une atteinte à l'intégrité. Vous pouvez ignorer ce message.

Messages d'arrêt

Après avoir émis une commande STOP QMGR, vous obtenez les messages CSQY009I et CSQY002I, par exemple:


```
CSQY009I +CSQ1 ' STOP QMGR' COMMAND ACCEPTED FROM  
USER(userid), STOP MODE(FORCE)  
CSQY002I +CSQ1 QUEUE MANAGER STOPPING
```

Où *userid* est l'ID utilisateur qui a émis la commande STOP QMGR et le paramètre MODE dépend de celui spécifié dans la commande.

Lorsque la commande STOP a abouti, les messages suivants s'affichent sur la console z/OS :

```
CSQ9022I +CSQ1 CSQYASCP ' STOP QMGR' NORMAL COMPLETION  
CSQ3104I +CSQ1 CSQ3EC0X - TERMINATION COMPLETE
```

Si vous utilisez ARM et que vous n'avez pas indiqué MODE (RESTART), le message suivant s'affiche également:

```
CSQY204I +CSQ1 ARM DEREGISTER for element arm-element type  
arm-element-type successful
```

Vous ne pouvez pas redémarrer le gestionnaire de files d'attente tant que le message suivant n'a pas été affiché:

```
CSQ3100I +CSQ1 CSQ3EC0X - SUBSYSTEM ssnm READY FOR START COMMAND
```



Panneaux d'opérations et de contrôle pour IBM MQ for z/OS

Vous pouvez utiliser les panneaux d'opérations et de contrôle IBM MQ pour effectuer des tâches d'administration sur les objets IBM MQ . Utilisez cette rubrique comme introduction aux commandes et aux panneaux de contrôle.

Ces panneaux permettent de définir, d'afficher, de modifier ou de supprimer des objets IBM MQ . Utilisez les panneaux pour l'administration quotidienne et pour apporter de petites modifications aux objets. Si vous configurez ou modifiez de nombreux objets, utilisez la fonction COMMAND du programme utilitaire CSQUTIL.

Les panneaux d'opérations et de contrôle prennent en charge les commandes de l'initiateur de canal (par exemple, pour démarrer un canal ou un programme d'écoute TCP/IP), pour la mise en cluster et pour la sécurité. Ils vous permettent également d'afficher des informations sur les unités d'exécution et l'utilisation des ensembles de pages.

Les panneaux fonctionnent en envoyant des commandes IBM MQ de type MQSC à un gestionnaire de files d'attente via la file d'attente d'entrée des commandes système.

Remarque :

1. Les panneaux d'opérations et de contrôle z/OS IBM MQ (CSQOREXX) peuvent ne pas prendre en charge toutes les nouvelles fonctions et tous les nouveaux paramètres ajoutés à partir de la version 7. Par exemple, il n'y a pas de panneaux pour la manipulation directe des objets de rubrique ou des abonnements.

L'utilisation de l'un des mécanismes pris en charge suivants vous permet d'administrer les définitions de publication / abonnement et d'autres contrôles système qui ne sont pas directement disponibles à partir d'autres panneaux:

- a. IBM MQ Explorer

- b. console z/OS
- c. Messages PCF (Programmable Command Format)
- d. Fonction COMMAND de CSQUTIL

Notez que l'action **Command** générique dans les panneaux CSQOREXX vous permet d'émettre n'importe quelle commande MQSC valide, y compris les commandes liées au fichier SMDS. Vous pouvez utiliser toutes les commandes émises par la fonction COMMAND de CSQUTIL.

2. Vous ne pouvez pas exécuter les commandes IBM MQ directement à partir de la ligne de commande dans les panneaux.
3. Pour utiliser les panneaux d'opérations et de contrôle, vous devez disposer de l'autorisation de sécurité appropriée ; elle est décrite dans la rubrique ID utilisateur pour la sécurité des commandes et la sécurité des ressources de commandes.
4. Vous ne pouvez pas indiquer d'ID utilisateur et de mot de passe à l'aide de CSQUTIL ou des panneaux CSQOREXX. A la place, si votre ID utilisateur dispose des droits UPDATE sur le profil BATCH dans MQCONN, vous pouvez ignorer le paramètre **CHCKLOCL**(*REQUIRED*) . Pour plus d'informations, voir Utilisation de CHCKLOCL sur des applications liées localement .

Appel et règles pour les panneaux d'opérations et de contrôle

Vous pouvez contrôler IBM MQ et émettre des commandes de contrôle via les panneaux ISPF.

Comment accéder aux panneaux d'opérations et de contrôle IBM MQ

Si le menu d'options principal ISPF/PDF a été mis à jour pour IBM MQ, vous pouvez accéder aux panneaux d'opérations et de contrôle IBM MQ à partir de ce menu. Pour plus de détails sur la mise à jour du menu, voir Tâche 20: Configurer les panneaux d'opérations et de contrôle.

Vous pouvez accéder aux panneaux d'opérations et de contrôle IBM MQ à partir du panneau du processeur de commandes TSO (généralement l'option 6 du menu principal des options ISPF/PDF). Le nom de la commande exec que vous exécutez pour effectuer cette opération est CSQOREXX. Il comporte deux paramètres: th1qua1 est le qualificatif de haut niveau pour les bibliothèques IBM MQ à utiliser et lang1letter est la lettre identifiant les bibliothèques de langue nationale à utiliser (par exemple, E pour U.S). Anglais). Les paramètres peuvent être omis si les bibliothèques IBM MQ sont installées de manière permanente dans votre configuration ISPF. Vous pouvez également exécuter CSQOREXX à partir de la ligne de commande TSO.

Ces panneaux sont conçus pour être utilisés par les opérateurs et les administrateurs avec un minimum de formation formelle. Lisez ces instructions avec les panneaux en cours d'exécution et essayez les différentes tâches suggérées.

Remarque : Lors de l'utilisation des panneaux, des files d'attente dynamiques temporaires avec des noms sous la forme SYSTEM.CSQOREXX.* sont créés.

Règles pour les panneaux d'opérations et de contrôle

Pour plus d'informations sur les règles générales relatives aux chaînes de caractères et aux noms IBM MQ , voir Règles de dénomination des objets IBM MQ . Toutefois, certaines règles s'appliquent uniquement aux panneaux d'opérations et de contrôle:

- Ne placez pas les chaînes, par exemple les descriptions, entre guillemets simples ou doubles.
- Si vous incluez une apostrophe ou un guillemet dans une zone de texte, vous n'avez pas besoin de la répéter ni d'ajouter un caractère d'échappement. Les caractères sont sauvegardés exactement au fur et à mesure que vous les saisissez ; par exemple:

This is Maria's queue

Le processeur de panneau les double pour que vous puissiez les transmettre à IBM MQ. Toutefois, s'il doit tronquer vos données pour ce faire, il le fait.

- Vous pouvez utiliser des caractères en majuscules ou en minuscules dans la plupart des zones, et ils sont réduits en majuscules lorsque vous appuyez sur Entrée. Voici les exceptions :
 - Noms de classe de stockage et noms de structure d'unité de couplage, qui doivent commencer par des majuscules de A à Z et être suivis par des majuscules de A à Z ou des caractères numériques.
 - Certaines zones ne sont pas traduites. Ces gestionnaires sont les suivants :
 - ID application
 - Description
 - Données d'environnement
 - Noms d'objet (mais si vous utilisez un nom d'objet en minuscules, il se peut que vous ne puissiez pas l'entrer dans une console z/OS)
 - Nom du système distant
 - Données de déclenchement
 - Données utilisateur
- Dans les noms, les blancs de début et les traits de soulignement de début sont ignorés. Par conséquent, vous ne pouvez pas avoir de noms d'objet commençant par des blancs ou des traits de soulignement.
- Les traits de soulignement sont utilisés pour afficher l'étendue des zones vides. Lorsque vous appuyez sur Entrée, les traits de soulignement de fin sont remplacés par des blancs.
- De nombreuses zones de description et de texte sont présentées en plusieurs parties, chaque partie étant gérée par IBM MQ de manière indépendante. Cela signifie que les blancs de fin sont conservés et que le texte n'est pas contigu.

Zones vides

Lorsque vous spécifiez l'action **Définir** pour un objet IBM MQ , chaque zone du panneau de définition contient une valeur. Pour plus d'informations sur l'emplacement où IBM MQ obtient les valeurs, voir l'aide générale (aide étendue) des panneaux d'affichage. Si vous remplacez une zone par des blancs et que ces derniers ne sont pas autorisés, IBM MQ place la valeur par défaut de l'installation dans la zone ou vous invite à entrer la valeur requise.

Lorsque vous spécifiez l'action **Modifier** pour un objet IBM MQ , chaque zone du panneau de modification contient la valeur en cours de cette zone. Si vous tapez des blancs dans une zone et que les blancs ne sont pas admis, la valeur de cette zone reste inchangée.

Objets et actions sur z/OS

Les panneaux d'opérations et de contrôle vous offrent de nombreux types d'objet différents et un certain nombre d'actions que vous pouvez effectuer sur eux.

Les actions sont répertoriées dans le panneau initial et vous permettent de manipuler les objets et d'afficher des informations les concernant. Ces objets incluent tous les objets IBM MQ , ainsi que certains objets supplémentaires. Les objets appartiennent aux catégories suivantes.

- Files d'attente, processus, objets d'informations d'authentification, listes de noms, classes de stockage et structures d'unité de couplage
- Canaux
- Objets de cluster
- Gestionnaire de files d'attente et sécurité
- Connexions
- Système

Voir **Actions** pour obtenir un tableau de références croisées des actions pouvant être exécutées avec les objets IBM MQ .

Files d'attente, processus, objets d'informations d'authentification, listes de noms, classes de stockage et structures d'unité de couplage

Il s'agit des objets IBM MQ de base. Il peut exister plusieurs éléments de chaque type. Ils peuvent être répertoriés, répertoriés avec un filtre, définis et supprimés, et avoir des attributs qui peuvent être affichés et modifiés, à l'aide des actions LIST ou DISPLAY, LIST avec FILTER, DEFINE LIKE, MANAGE et ALTER. (Les objets sont supprimés à l'aide de l'action MANAGE.)

Cette catégorie comprend les objets suivants:

| | |
|-----------|--|
| QLOCAL | File d'attente locale |
| QREMOTE | File d'attente éloignée |
| QALIAS | File d'attente alias pour la référence indirecte à une file d'attente |
| QMODEL | File d'attente modèle pour la définition dynamique de files d'attente |
| QUEUE | Tout type de file d'attente |
| QSTATUS | Statut d'une file d'attente locale |
| PROCESS | Informations sur une application à démarrer lors d'un événement de déclenchement |
| AUTHINFO | Informations d'authentification : définitions requises pour l'exécution de la vérification par rapport à la liste de retrait des certificats (CRL) à l'aide de serveurs LDAP |
| NAMELIST | Liste de noms, telles des files d'attente ou des clusters |
| STGCLASS | Classe d'archivage |
| CFSTRUCT | Structure de l'unité de couplage |
| STATUT FC | Statut d'une structure d'unité de couplage |

Canaux

Les canaux sont utilisés pour la mise en file d'attente répartie. Il peut y avoir beaucoup de chaque type, et ils peuvent être listés, listés avec filtre, définis, supprimés, affichés et modifiés. D'autres fonctions sont également disponibles à l'aide des actions START, STOP et PERFORM. PERFORM fournit des fonctions de réinitialisation, de commande ping et de résolution de canal.

Cette catégorie comprend les objets suivants:

| | |
|-----------|---------------------------------|
| Canal | Tout type de canal |
| EMETTEUR | Canal émetteur |
| SERVEUR | Canal serveur |
| RECEPTEUR | Canal récepteur |
| DEMANDEUR | Canal demandeur |
| CLUSRCVR | Canal récepteur de cluster |
| CLUSSDR | Canal émetteur de cluster |
| SVRCONN | Canal de connexion serveur |
| CLNTCONN | Canal de connexion client |
| STATUT | Statut d'une connexion de canal |

Objets de cluster

Les objets de cluster sont créés automatiquement pour les files d'attente et les canaux appartenant à un cluster. La file d'attente de base et les définitions de canal peuvent se trouver sur un autre gestionnaire de files d'attente. Il peut exister plusieurs éléments de chaque type et les noms en double sont admis. Ils peuvent être répertoriés, répertoriés avec un filtre et affichés. PERFORM, START et STOP sont également disponibles via les actions LIST.

Cette catégorie comprend les objets suivants:

| | |
|----------|---|
| CLUSQ | File d'attente de cluster, créée pour une file d'attente appartenant à un cluster |
| CLUSCHL | Canal de cluster, créé pour un canal appartenant à un cluster |
| CLUSQMGR | Gestionnaire de files d'attente de cluster, identique à un canal de cluster mais identifié par son nom de gestionnaire de files d'attente |

Les canaux de cluster et les gestionnaires de files d'attente de cluster possèdent les actions PERFORM, START et STOP, mais uniquement indirectement via l'action DISPLAY.

Gestionnaire de files d'attente et sécurité

Les objets de gestionnaire de files d'attente et de sécurité ont une seule instance. Ils peuvent être répertoriés et comporter des attributs qui peuvent être affichés et modifiés (à l'aide des actions LIST ou DISPLAY et ALTER), ainsi que d'autres fonctions disponibles à l'aide de l'action PERFORM.

Cette catégorie comprend les objets suivants:

| | |
|--------------|---|
| GESTIONNAIRE | Gestionnaire de files d'attente: l'action PERFORM fournit des fonctions d'interruption et de reprise de cluster |
| Sécurité | Fonctions de sécurité: l'action PERFORM fournit des fonctions d'actualisation et de revérification |

Connexion

Les connexions peuvent être répertoriées, répertoriées avec un filtre et affichées.

Cette catégorie se compose uniquement de l'objet de connexion CONNECT.

Système

Ensemble d'autres fonctions. Cette catégorie comprend les objets suivants:

| | |
|---------|--------------------|
| SYSTEME | Fonctions système |
| CONTROL | Synonyme de SYSTEM |

Les fonctions disponibles sont les suivantes :

| | |
|-----------------|---|
| LIST ou DISPLAY | Afficher les informations d'utilisation du groupe de partage de files d'attente, de la mise en file d'attente répartie, de l'ensemble de pages ou du fichier. |
| exécuter | Actualiser ou réinitialiser la mise en cluster |
| DEBUT | Démarrer l'initiateur de canal ou les programmes d'écoute |
| ARRETER | Arrêter l'initiateur de canal ou les programmes d'écoute |

Actions

Les actions que vous pouvez effectuer pour chaque type d'objet sont présentées dans le tableau suivant:

Tableau 25. Opérations valides et actions du panneau de commande pour les objets IBM MQ

| Objet | Modifier | Définir comme | Gérer (1) | Liste ou affichage | Liste avec filtre | exécuter | Démarrer | Arrêter |
|--------------|----------|---------------|-----------|--------------------|-------------------|----------|----------|---------|
| AUTHINFO | X | X | X | X | X | | | |
| STATUT FC | | | | X | | | | |
| CFSTRUCT | X | X | X | X | X | | | |
| Canal | X | X | X | X | X | X | X | X |
| STATUT | | | | X | X | | | |
| CLNTCONN | X | X | X | X | X | | | |
| CLUSCHL | | | | X | X | X (2) | X (2) | X (2) |
| CLUSQ | | | | X | X | | | |
| CLUSQMGR | | | | X | X | X (2) | X (2) | X (2) |
| CLUSRCVR | X | X | X | X | X | X | X | X |
| CLUSDR | X | X | X | X | X | X | X | X |
| Se connecter | | | | X | X | | | |
| CONTROL | | | | X | | X | X | X |
| GESTIONNAIRE | X | | | X | | X | | |
| NAMELIST | X | X | X | X | X | | | |
| PROCESS | X | X | X | X | X | | | |
| QALIAS | X | X | X | X | X | | | |
| QLOCAL | X | X | X | X | X | | | |
| QMODEL | X | X | X | X | X | | | |
| QREMOTE | X | X | X | X | X | | | |
| QSTATUS | | | | X | X | | | |
| QUEUE | X | X | X | X | X | | | |
| RECEPTEUR | X | X | X | X | X | X | X | X |
| DEMANDEUR | X | X | X | X | X | X | X | X |
| Sécurité | X | | | X | | X | | |
| EMETTEUR | X | X | X | X | X | X | X | X |
| SERVEUR | X | X | X | X | X | X | X | X |
| SVRCONN | X | X | X | X | X | | X | X |
| STGCLASS | X | X | X | X | X | | | |
| SYSTEME | | | | X | | X | X | X |

Remarque :

1. Fournit Supprimer et d'autres fonctions

2. Utilisation de l'action Liste ou affichage

Dispositions d'objet sur z/OS

Vous pouvez spécifier la *disposition* de l'objet avec lequel vous devez travailler. La disposition indique où la **définition** de l'objet est conservée et comment l'objet se comporte.

La disposition est significative uniquement si vous utilisez l'un des types d'objet suivants:

- queues
- canaux
- processus
- liste de noms
- classe d'archivage
- objets d'informations d'authentification

Si vous utilisez d'autres types d'objet, la disposition est ignorée.

Les valeurs admises sont les suivantes:

Q

QMGR. Les définitions d'objet se trouvent dans l'ensemble de pages du gestionnaire de files d'attente et sont accessibles uniquement par le gestionnaire de files d'attente.

C

COPY (Copier). Les définitions d'objet se trouvent dans l'ensemble de pages du gestionnaire de files d'attente et sont accessibles uniquement par le gestionnaire de files d'attente. Il s'agit de copies locales d'objets définis comme ayant une disposition de GROUP.

P

PRIVÉ. Les définitions d'objet se trouvent dans l'ensemble de pages du gestionnaire de files d'attente et sont accessibles uniquement par le gestionnaire de files d'attente. Les objets ont été définis comme ayant une disposition de QMGR ou COPY.

G

. Les définitions d'objet se trouvent dans le référentiel partagé et sont accessibles à tous les gestionnaires de files d'attente du groupe de partage de files d'attente.

S

Partagé. Cette disposition s'applique uniquement aux files d'attente locales. Les définitions de file d'attente se trouvent dans le référentiel partagé et sont accessibles à tous les gestionnaires de files d'attente du groupe de partage de files d'attente.

A

TOUS. Si le gestionnaire de files d'attente d'action est le gestionnaire de files d'attente cible ou *, les objets de **toutes** les dispositions sont inclus ; dans le cas contraire, les objets des dispositions QMGR et COPY uniquement sont inclus. Il s'agit de l'option par défaut.

Sélection d'un gestionnaire de files d'attente, de valeurs par défaut et de niveaux à l'aide du panneau de commande ISPF sous z/OS

Vous pouvez utiliser la commande exec CSQOREXX dans ISPF pour contrôler vos gestionnaires de files d'attente.

Lorsque vous visualisez le panneau initial, vous n'êtes connecté à aucun gestionnaire de files d'attente. Toutefois, dès que vous appuyez sur Entrée, vous êtes connecté au gestionnaire de files d'attente ou à un gestionnaire de files d'attente du groupe de partage de files d'attente nommé dans la zone **Nom de connexion**. Vous pouvez laisser cette zone vide ; cela signifie que vous utilisez le gestionnaire de files d'attente par défaut pour les applications par lots. Ceci est défini dans CSQBDEFV (voir [Tâche 19: Configurer les adaptateurs Batch, TSO et RRS pour plus d'informations à ce sujet](#)).

Utilisez la zone **Gestionnaire de files d'attente cible** pour spécifier le gestionnaire de files d'attente dans lequel les actions que vous demandez doivent être effectuées. Si vous ne renseignez pas cette zone, le gestionnaire de files d'attente spécifié dans la zone **Nom de connexion** est utilisé par défaut. Vous pouvez

spécifier un gestionnaire de files d'attente cible qui n'est pas celui auquel vous vous connectez. Dans ce cas, vous devez normalement indiquer le nom d'un objet de gestionnaire de files d'attente éloignées qui fournit une définition d'alias de gestionnaire de files d'attente (ce nom est utilisé comme *ObjectQMgrNom* lors de l'ouverture de la file d'attente d'entrée de commandes). Pour ce faire, vous devez disposer de files d'attente et de canaux appropriés configurés pour accéder au gestionnaire de files d'attente éloignées.

La zone **Gestionnaire de files d'attente d'action** permet de spécifier un gestionnaire de files d'attente qui se trouve dans le même groupe de partage de files d'attente que le gestionnaire de files d'attente spécifié dans la zone **Gestionnaire de files d'attente cible** comme gestionnaire de files d'attente dans lequel les actions que vous demandez doivent être exécutées. Si vous indiquez * dans cette zone, les actions que vous demandez sont effectuées sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Si vous laissez cette zone vide, elle prend par défaut la valeur spécifiée dans la zone **Gestionnaire de files d'attente cible**. La zone **Gestionnaire de files d'attente d'actions** correspond à l'utilisation du modificateur de commande CMDSCOPE décrit dans [Les commandes MQSC](#).

Valeurs par défaut du gestionnaire de files

Si vous laissez les zones de gestionnaire de files d'attente vides ou que vous choisissez de vous connecter à un groupe de partage de files d'attente, une fenêtre secondaire s'ouvre lorsque vous appuyez sur **Entrée**. Cette fenêtre confirme les noms des gestionnaires de files d'attente que vous utiliserez. Appuyez sur **Entrée** pour continuer. Lorsque vous revenez au panneau initial après avoir effectué des demandes, vous trouvez les zones renseignées avec les noms réels.

Niveaux de gestionnaire de files d'attente

Les panneaux d'opérations et de contrôle fonctionnent de manière satisfaisante uniquement avec les gestionnaires de files d'attente s'exécutant sous z/OS à la version IBM WebSphere MQ 710 ou ultérieure.

Si ces conditions ne sont pas remplies, il est probable que les actions ne fonctionnent que partiellement, de manière incorrecte ou pas du tout et que les réponses du gestionnaire de files d'attente ne sont pas reconnues.

Si le gestionnaire de files d'attente d'action n'est pas au niveau ou au-dessus de IBM MQ 8.0.0, certaines zones ne sont pas affichées et certaines valeurs ne peuvent pas être entrées. Quelques objets et actions sont interdits. Dans ce cas, une fenêtre secondaire s'ouvre pour vous demander de confirmer que vous souhaitez continuer.

Utilisation des touches de fonction et de la ligne de commande avec les panneaux de contrôle ISPF sous z/OS

Pour utiliser les panneaux, vous devez utiliser les touches de fonction ou entrer les commandes équivalentes dans la zone de commande du panneau de commande ISPF.

- [Touches de fonction](#)
 - [Traitement de vos actions](#)
 - [«Affichage des messages utilisateur IBM MQ», à la page 361](#)
 - [Annulation de vos actions](#)
 - [Obtenir de l'aide](#)
- [Utilisation de la ligne de commande](#)

Touches de fonction

Les touches de fonction possèdent des paramètres spéciaux pour IBM MQ. (Cela signifie que vous ne pouvez pas utiliser les valeurs par défaut ISPF pour les touches de fonction ; si vous avez déjà utilisé la commande KEYLIST OFF ISPF n'importe où, vous devez entrer KEYLIST ON dans la zone de commande d'un panneau d'opérations et de contrôle, puis appuyer sur Entrée pour activer les paramètres IBM MQ.)

Ces paramètres de touche de fonction peuvent être affichés sur les panneaux, comme illustré dans la Figure 45, à la page 362. Si les paramètres ne sont pas affichés, entrez PFSHOW dans la zone de commande d'un panneau d'opérations et de contrôle, puis appuyez sur **Entrée**. Pour supprimer l'affichage des paramètres, utilisez la commande PFSHOW OFF.

Les paramètres des touches de fonction dans les panneaux d'opérations et de contrôle sont conformes aux normes CUA. Bien que vous puissiez modifier le paramètre de clé via des procédures ISPF normales (telles que l'utilitaire **KEYLIST**), il n'est pas recommandé de le faire.

Remarque : L'utilisation des commandes **PFSHOW** et **KEYLIST** affecte tous les autres écrans ISPF logiques dont vous disposez et leurs paramètres sont conservés lorsque vous quittez les panneaux d'opérations et de contrôle.

Traitement de vos actions

Appuyez sur **Entrée** pour effectuer l'action demandée sur un panneau. Les informations du panneau sont envoyées au gestionnaire de files d'attente pour traitement.

Chaque fois que vous appuyez sur **Entrée** dans les panneaux, IBM MQ génère un ou plusieurs messages opérateur. Si l'opération a abouti, vous obtenez le message de confirmation CSQ9022I, sinon vous obtenez des messages d'erreur.

Affichage des messages utilisateur IBM MQ

Appuyez sur la touche de fonction F10 dans n'importe quel panneau pour afficher les messages utilisateur IBM MQ.

Annulation de vos actions

Sur le panneau initial, F3 et F12 quittent les panneaux d'opérations et de contrôle et vous renvoie à ISPF. Aucune information n'est envoyée au gestionnaire de files d'attente.

Sur tout autre panneau, appuyez sur les touches de fonction F3 ou F12 pour quitter le panneau en cours **en ignorant les données que vous avez saisies depuis le dernier appui sur Entrée**. A nouveau, aucune information n'est envoyée au gestionnaire de files d'attente.

- F3 vous ramène directement au panneau initial.
- F12 vous ramène au panneau précédent.

Obtenir de l'aide

Chaque panneau est associé à des panneaux d'aide. Les panneaux d'aide utilisent les protocoles ISPF:

- Appuyez sur la touche de fonction F1 dans n'importe quel panneau pour afficher l'aide générale (aide étendue) sur la tâche.
- Appuyez sur la touche de fonction F1 avec le curseur sur n'importe quelle zone pour afficher une aide spécifique sur cette zone.
- Appuyez sur la touche de fonction F5 à partir de n'importe quel panneau d'aide de zone pour obtenir l'aide générale.
- Appuyez sur la touche de fonction F3 pour revenir au panneau de base, c'est-à-dire le panneau à partir duquel vous avez appuyé sur la touche de fonction F1.
- Appuyez sur la touche de fonction F6 à partir de n'importe quel panneau d'aide pour obtenir de l'aide sur les touches de fonction.

Si les informations d'aide se trouvent dans une deuxième page ou dans les pages suivantes, un indicateur **Plus** s'affiche dans l'angle supérieur droit du panneau. Utilisez ces touches de fonction pour naviguer dans les pages d'aide:

- F11 pour accéder à la page d'aide suivante (le cas échéant).
- F10 pour revenir à la page d'aide précédente (le cas échéant).

Utilisation de la ligne de commande

Vous n'avez jamais besoin d'utiliser la ligne de commande pour exécuter les commandes utilisées par les panneaux d'opérations et de contrôle car elles sont disponibles à partir des touches de fonction. La ligne de commande est fournie pour vous permettre d'entrer des commandes ISPF normales (telles que **PFSHOW**).

La commande ISPF PANELID ON affiche le nom du panneau CSQOREXX en cours.

La ligne de commande s'affiche initialement dans la position par défaut au bas des panneaux, quels que soient les paramètres ISPF dont vous disposez. Vous pouvez utiliser la commande SETTINGS ISPF à partir de n'importe quel panneau d'opérations et de contrôle pour modifier la position de la ligne de commande. Les paramètres sont mémorisés pour les sessions suivantes avec les panneaux d'opérations et de contrôle.

Utilisation des panneaux d'opérations et de contrôle sous z/OS

Utilisez cette rubrique pour examiner le panneau de commande initial affiché à partir de CSQOREXX

La [Figure 45](#), à la page 362 montre le panneau qui s'affiche lorsque vous démarrez une session de panneau.

```

IBM MQ for z/OS - Main Menu
Complete fields. Then press Enter.
Action . . . . . 1      0. List with filter  4. Manage
                        1. List or Display  5. Perform
                        2. Define like    6. Start
                        3. Alter          7. Stop
                        8. Command
Object type . . . . . CHANNEL +
Name . . . . . *
Disposition . . . . . A  Q=Qmgr, C=Copy, P=Private, G=Group,
                        S=Shared, A=All

Connect name . . . . . MQ1C - local queue manager or group
Target queue manager . . . MQ1C
                        - connected or remote queue manager for command input
Action queue manager . . . MQ1C - command scope in group
Response wait time . . . . 30 5 - 999 seconds

(C) Copyright IBM Corporation 1993, 2023. All rights reserved.

Command ==>
F1=Help      F2=Split    F3=Exit     F4=Prompt   F9=SwapNext F10=Messages
F12=Cancel

```

Figure 45. Panneau IBM MQ d'opérations et de contrôle initial

A partir de ce panneau, vous pouvez effectuer des actions telles que:

- Choisissez le gestionnaire de files d'attente local de votre choix et indiquez si vous souhaitez que les commandes soient émises sur ce gestionnaire de files d'attente, sur un gestionnaire de files d'attente éloignées ou sur un autre gestionnaire de files d'attente du même groupe de partage de files d'attente que le gestionnaire de files d'attente local. Remplacez le nom du gestionnaire de files d'attente si vous devez le modifier.
- Sélectionnez l'action que vous souhaitez effectuer en entrant le numéro approprié dans la zone **Action**.
- Indiquez le type d'objet à utiliser. Appuyez sur la touche de fonction F1 pour obtenir de l'aide sur les types d'objet si vous n'êtes pas sûr de leur nature.
- Indiquez la disposition du type d'objet que vous souhaitez utiliser.
- Affiche la liste des objets du type indiqué. Entrez un astérisque (*) dans la zone **Nom** et appuyez sur **Entrée** pour afficher la liste des objets (du type spécifié) qui ont déjà été définis sur le gestionnaire de files d'attente d'action. Vous pouvez ensuite sélectionner un ou plusieurs objets à utiliser en séquence. Toutes les actions sont disponibles dans la liste.

Remarque : Il est recommandé d'effectuer des choix qui entraînent l'affichage d'une liste d'objets, puis de travailler à partir de cette liste. Utilisez l'action **Afficher**, car elle est autorisée pour tous les types d'objet.

Utilisation de la fonction de commande sous z/OS

Utilisez l'éditeur pour entrer ou modifier les commandes MQSC à transmettre au gestionnaire de files d'attente.

Dans le panneau principal, CSQOPRIA, sélectionnez l'option **8 Command** pour démarrer la fonction de commande.

Une session d'édition d'un fichier séquentiel, *préfixe*, s'affiche. CSQUTIL.COMMANDS, utilisé comme entrée de la fonction CSQUTIL COMMAND ; voir [Emission de commandes pour IBM MQ](#).

Vous n'avez pas besoin de préfixer les commandes avec la chaîne de préfixe de commande (CPF).

Vous pouvez continuer les commandes MQSC sur les lignes suivantes en terminant la ligne en cours par les caractères de continuation + ou -. Vous pouvez également utiliser le mode d'édition de ligne pour fournir des commandes MQSC longues ou les valeurs des valeurs d'attribut longues dans la commande.

édition de ligne

Pour utiliser l'édition de ligne, déplacez le curseur sur la ligne appropriée dans le panneau d'édition et utilisez **F4** pour afficher une seule ligne dans un panneau de défilement. Une seule ligne peut contenir jusqu'à 32 760 octets de données.

Pour quitter l'édition de ligne:

- **F3 exit** enregistre les modifications apportées à la ligne et quitte
- **F12 cancel** revient au panneau d'édition, qui annule les modifications apportées à la ligne.

Pour annuler les modifications apportées dans la session d'édition, utilisez **F12 cancel** pour mettre fin à la session d'édition en laissant le contenu du fichier inchangé. Les commandes ne sont pas exécutées.

Exécution des commandes

Une fois que vous avez terminé d'entrer les commandes MQSC, mettez fin à la session d'édition avec **F3 exit** pour sauvegarder le contenu du fichier et appelez CSQUTIL pour transmettre les commandes au gestionnaire de files d'attente. La sortie du traitement des commandes est conservée dans le fichier *préfixe*.CSQUTIL.OUTPUT. Une session d'édition s'ouvre automatiquement sur ce fichier pour que vous puissiez afficher les réponses. Appuyez sur **F3 exit** pour quitter cette session et revenir au menu principal.

Utilisation des objets IBM MQ sous z/OS

La plupart des tâches décrites dans cette documentation impliquent la manipulation d'objets IBM MQ . Les types d'objet sont les gestionnaires de files d'attente, les files d'attente, les définitions de processus, les listes de noms, les canaux, les canaux de connexion client, les programmes d'écoute, les services et les objets d'informations d'authentification.

- [Définition d'objets file d'attente simples](#)
- [Définition d'autres types d'objet](#)
- [Utilisation des définitions d'objet](#)
- [Utilisation des listes de noms](#)

Définition d'objets file d'attente simples

Pour définir un nouvel objet, utilisez une définition existante comme base. Vous pouvez effectuer cette opération de l'une des trois manières suivantes:

- En sélectionnant un objet qui est membre d'une liste affichée suite aux options sélectionnées sur le panneau initial. Vous entrez ensuite le type d'action 2 (**Définir comme**) dans la zone d'action en regard de l'objet sélectionné. Votre nouvel objet possède les attributs de l'objet sélectionné, à l'exception de la disposition. Vous pouvez ensuite modifier les attributs de votre nouvel objet en fonction de vos besoins.
- Dans le panneau initial, sélectionnez le type d'action **Définir comme** , entrez le type d'objet que vous définissez dans la zone **Type d'objet** et entrez le nom d'un objet existant spécifique dans la zone **Nom** . Votre nouvel objet possède les mêmes attributs que l'objet que vous avez nommé dans la zone **Nom** , à l'exception de la disposition. Vous pouvez ensuite modifier les attributs de votre nouvelle définition d'objet en fonction de vos besoins.

- En sélectionnant le type d'action **Définir comme** , en spécifiant un type d'objet, puis en laissant la zone **Nom** vide. Vous pouvez ensuite définir votre nouvel objet et il possède les attributs par défaut définis pour votre installation. Vous pouvez ensuite modifier les attributs de votre nouvelle définition d'objet en fonction de vos besoins.

Remarque : Vous n'entrez pas le nom de l'objet que vous définissez dans le panneau initial, mais dans le panneau **Définir** qui vous est présenté.

L'exemple suivant montre comment définir une file d'attente locale en utilisant une file d'attente existante comme modèle.

Définition d'une file d'attente locale

Pour définir un objet de file d'attente locale à partir des panneaux d'opérations et de contrôle, utilisez une définition de file d'attente existante comme base de votre nouvelle définition. Il y a plusieurs panneaux à compléter. Une fois que vous avez rempli tous les panneaux et que vous avez vérifié que les attributs sont corrects, appuyez sur Entrée pour envoyer votre définition au gestionnaire de files d'attente, qui crée ensuite la file d'attente réelle.

Utilisez l'action **Définir comme** sur le panneau initial ou sur une entrée d'objet dans une liste affichée suite aux options sélectionnées sur le panneau initial.

Par exemple, à partir du panneau initial, renseignez les zones suivantes:

| | |
|---------------------|---|
| Action | 2 (Définir comme) |
| Type d'objet | QLOCAL |
| Nom | QUEUE.YOU.LIKE. Il s'agit du nom de la file d'attente qui fournit les attributs de votre nouvelle file d'attente. |

Appuyez sur Entrée pour afficher le panneau **Definie a Local Queue** . La zone du nom de la file d'attente est vide pour que vous puissiez indiquer le nom de la nouvelle file d'attente. La description est celle de la file d'attente sur laquelle vous basez cette nouvelle définition. Entrez cette zone avec votre propre description pour la nouvelle file d'attente.

Les valeurs des autres zones sont celles de la file d'attente sur laquelle vous basez cette nouvelle file d'attente, à l'exception de la disposition. Vous pouvez sur-taper ces zones en fonction de vos besoins. Par exemple, entrez Y dans la zone **Insertion activée** (si elle n'est pas déjà Y) si les applications disposant des droits appropriés peuvent insérer des messages dans cette file d'attente.

Pour obtenir de l'aide sur les zones, déplacez le curseur dans une zone et appuyez sur la touche de fonction F1. L'aide sur les zones fournit des informations sur les valeurs pouvant être utilisées pour chaque attribut.

Une fois le premier panneau terminé, appuyez sur la touche de fonction F8 pour afficher le second panneau.

Astuces :

1. N'appuyez pas sur Entrée à ce stade, sinon la file d'attente sera créée avant que vous ayez la possibilité de renseigner les zones restantes. (Si vous appuyez sur Entrée prématurément, ne vous inquiétez pas ; vous pouvez toujours modifier votre définition ultérieurement.)
2. N'appuyez pas sur les touches de fonction F3 ou F12, sinon les données que vous avez saisies seront perdues.

Appuyez à plusieurs reprises sur la touche de fonction F8 pour afficher et compléter les panneaux restants, notamment les panneaux de définition de déclencheur, de contrôle d'événement et d'annulation de rapport.

Lorsque votre définition de file d'attente locale est terminée

Une fois la définition terminée, appuyez sur Entrée pour envoyer les informations au gestionnaire de files d'attente en vue de leur traitement. Le gestionnaire de files d'attente crée la file d'attente en

fonction de la définition que vous avez fournie. Si vous ne souhaitez pas que la file d'attente soit créée, appuyez sur la touche de fonction F3 pour quitter et annuler la définition.

Définition d'autres types d'objet

Pour définir d'autres types d'objet, utilisez une définition existante comme base de votre nouvelle définition, comme expliqué dans [Définition d'une file d'attente locale](#).

Utilisez l'action **Définir comme** sur le panneau initial ou sur une entrée d'objet dans une liste affichée suite aux options sélectionnées sur le panneau initial.

Par exemple, à partir du panneau initial, renseignez les zones suivantes:

| | |
|---------------------|--|
| Action | 2 (Définir comme) |
| Type d'objet | QALIAS, NAMELIST, PROCESS, CHANNEL et d'autres objets de ressource. |
| Nom | Laissez la zone à blanc ou entrez le nom d'un objet existant du même type. |

Appuyez sur Entrée pour afficher les panneaux DEFINE correspondants. Renseignez les zones requises, puis appuyez à nouveau sur Entrée pour envoyer les informations au gestionnaire de files d'attente.

A l'instar de la définition d'une file d'attente locale, la définition d'un autre type d'objet nécessite généralement la complétion de plusieurs panneaux. La définition d'une liste de noms nécessite un travail supplémentaire, comme décrit dans [«Utilisation des listes de noms»](#), à la page 366.

Utilisation des définitions d'objet

Lorsqu'un objet a été défini, vous pouvez spécifier une action dans la zone **Action** pour le modifier, l'afficher ou la gérer.

Dans chaque cas, vous pouvez:

- Sélectionnez l'objet à utiliser dans une liste affichée suite aux options sélectionnées dans le panneau initial. Par exemple, si vous avez entré 1 dans la zone **Action** pour afficher des objets, Queue dans la zone **Type d'objet** et * dans la zone **Nom**, la liste de toutes les files d'attente définies dans le système s'affiche. Vous pouvez ensuite sélectionner dans cette liste la file d'attente avec laquelle vous devez travailler.
- Commencez à partir du panneau initial, dans lequel vous spécifiez l'objet que vous utilisez en renseignant les zones **Type d'objet** et **Nom**.

Modification d'une définition d'objet

Pour modifier une définition d'objet, indiquez l'action 3 et appuyez sur Entrée pour afficher les panneaux ALTER. Ces panneaux sont très similaires aux panneaux DEFINE. Vous pouvez modifier les valeurs de votre choix. Lorsque vos modifications sont terminées, appuyez sur Entrée pour envoyer les informations au gestionnaire de files d'attente.

Affichage d'une définition d'objet

Si vous souhaitez afficher les détails d'un objet sans pouvoir les modifier, indiquez l'action 1 et appuyez sur Entrée pour afficher les panneaux DISPLAY. Là encore, ces panneaux sont similaires aux panneaux DEFINE, à ceci près que vous ne pouvez modifier aucune des zones. Modifiez le nom de l'objet pour afficher les détails d'un autre objet.

Suppression d'un objet

Pour supprimer un objet, spécifiez l'action 4 (Gérer) et l'action **Supprimer** est l'une des actions présentées dans le menu qui s'affiche. Sélectionnez l'action **Supprimer**.

Vous êtes invité à confirmer votre demande. Si vous appuyez sur la touche de fonction F3 ou F12, la demande est annulée. Si vous appuyez sur Entrée, la demande est confirmée et transmise au gestionnaire de files d'attente. L'objet que vous avez spécifié est ensuite supprimé.

Remarque : Vous ne pouvez pas supprimer la plupart des types d'objet canal sauf si l'initiateur de canal est démarré.

Utilisation des listes de noms

Lorsque vous utilisez des listes de noms, procédez comme vous le feriez pour d'autres objets.

Pour les actions DEFINE LIKE ou ALTER, appuyez sur la touche de fonction F11 pour ajouter des noms à la liste ou pour modifier les noms dans la liste. Cela implique l'utilisation de l'éditeur ISPF et toutes les commandes d'édition ISPF normales sont disponibles. Entrez chaque nom dans la liste de noms sur une ligne distincte.

Lorsque vous utilisez l'éditeur ISPF de cette manière, les paramètres des touches de fonction sont les paramètres ISPF normaux et **non** ceux utilisés par les autres panneaux d'opérations et de contrôle.

Si vous devez spécifier des noms en minuscules dans la liste, spécifiez CAPS (OFF) sur la ligne de commande du panneau de l'éditeur. Lorsque vous effectuez cette opération, toutes les listes de noms que vous éditez ultérieurement sont en minuscules jusqu'à ce que vous définissiez CAPS (ON).

Lorsque vous avez terminé d'éditer la liste de noms, appuyez sur la touche de fonction F3 pour mettre fin à la session d'édition ISPF. Appuyez ensuite sur Entrée pour envoyer les modifications au gestionnaire de files d'attente.

Avertissement : Si vous n'appuyez pas sur Entrée à ce stade mais que vous appuyez sur la touche de fonction F3 à la place, vous perdez toutes les mises à jour que vous avez saisies.

Implémentation du système à l'aide de plusieurs files d'attente de transmission de cluster

Cela ne fait aucune différence si le canal est utilisé dans un cluster unique ou dans un cluster qui se chevauche. Lorsque le canal est sélectionné et démarré, il sélectionne la file d'attente de transmission en fonction des définitions.

Procédure

- Si vous utilisez l'option DEFCLXQ, voir [«Utilisation de la définition automatique des files d'attente et commutation»](#), à la page 366.
- Si vous utilisez une approche par étapes, voir [«Modification de vos canaux émetteurs de cluster à l'aide d'une approche par étapes»](#), à la page 367.

Utilisation de la définition automatique des files d'attente et commutation

Utilisez cette option si vous prévoyez d'utiliser l'option DEFCLXQ. Une file d'attente sera créée pour chaque canal et chaque nouveau canal.

Procédure

1. Passez en revue la définition de SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE et modifiez les attributs si nécessaire.
Cette file d'attente est définie dans le membre SCSQPROC (csq4insx).
2. Créez le système SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE .
3. Appliquez des règles de sécurité pour cette file d'attente modèle et pour SYSTEM.CLUSTER.TRANSMIT.* * files d'attente.

Pour z/OS , l'ID utilisateur de la tâche démarrée de l'initiateur de canal doit être:

- Contrôle de l'accès à CLASS (MQADMIN) pour

```
ssid.CONTEXT.SYSTEM.CLUSTER.TRANSMIT.channelName
```

- Accès en mise à jour à CLASS (MQQUEUE) pour

```
ssid.SYSTEM.CLUSTER.TRANSMIT.channelname
```

Modification de vos canaux émetteurs de cluster à l'aide d'une approche par étapes

Utilisez cette option si vous prévoyez d'utiliser une approche par étapes. Ce processus vous permet de passer aux nouveaux canaux émetteurs de cluster à différents moments pour répondre aux besoins de votre entreprise.

Avant de commencer

- Identifiez vos applications métier et les canaux utilisés.
- Pour les files d'attente que vous utilisez, affichez les clusters dans lesquels elles se trouvent.
- Affichez les canaux pour afficher les noms de connexion, les noms des gestionnaires de files d'attente éloignées et les clusters pris en charge par le canal.

Pourquoi et quand exécuter cette tâche

- Créez une file d'attente de transmission. Sous z/OS, vous souhaitez peut-être prendre en compte l'ensemble de pages que vous utilisez pour la file d'attente.
- Configurez la règle de sécurité pour la file d'attente.
- Modifiez toute surveillance de file d'attente pour inclure ce nom de file d'attente.
- Choisissez les canaux qui doivent utiliser cette file d'attente de transmission. Les canaux doivent avoir un nom similaire, de sorte que les caractères génériques ' * ' dans le CLCHNAME, identifiez le canal.
- Lorsque vous êtes prêt à utiliser la nouvelle fonction, modifiez la file d'attente de transmission pour indiquer le nom des canaux devant utiliser cette file d'attente de transmission. Par exemple, CLUSTER1.TOPARIS ou CLUSTER1.* ou *.TOPARIS
- Démarrer les canaux

Procédure

1. La commande `DIS CLUSQMGR(xxxx) XMITQ` permet d'afficher les canaux émetteurs de cluster définis dans le cluster, où `xxxx` est le nom du gestionnaire de files d'attente éloignées.
2. Configurez le profil de sécurité de la file d'attente de transmission et accordez à la file d'attente l'accès à l'initiateur de canal.
3. Définissez la file d'attente de transmission à utiliser et indiquez `USAGE (XMITQ) INDXTYPE (CORRELID) SHARE` et `CLCHNAME (valeur)`

L'ID utilisateur de la tâche démarrée de l'initiateur de canal a besoin de l'accès suivant:

```
alter class(MQADMIN) ssid.CONTEXT.SYSTEM.CLUSTER.TRANSMIT.channel  
update class(MQQUEUE ssid.SYSTEM.CLUSTER.TRANSMIT.channel
```

et l'ID utilisateur à l'aide de la commande `SWITCH` a besoin de l'accès suivant:

```
alter cl(MQADMIN) ssid.QUEUE.queueaname
```

4. Arrêtez et redémarrez les canaux.

Le changement de canal se produit lorsque le canal démarre à l'aide d'une commande `MQSC` ou que vous utilisez `CSQUTIL`. Vous pouvez identifier les canaux qui doivent être redémarrés à l'aide du `SWITCH CHANNEL (*) STATUS` de `CSQUTIL`

Si vous rencontrez des problèmes lors du démarrage du canal, arrêtez le canal, résolvez les problèmes et redémarrez le canal.

Notez que vous pouvez modifier l'attribut CLCHNAME aussi souvent que nécessaire.

La valeur de CLCHNAME utilisée est celle utilisée lorsque le canal est démarré. Vous pouvez donc modifier la définition de CLCHNAME alors que le canal continue d'utiliser les définitions à partir du moment où il a été démarré. Le canal utilise la nouvelle définition lorsqu'il est redémarré.

Annulation d'une modification sur z/OS

Vous devez disposer d'un processus pour annuler une modification si les résultats ne sont pas ceux attendus.

Qu'est-ce qui peut mal tourner?

Si la nouvelle file d'attente de transmission n'est pas celle que vous attendez:

1. Vérifiez que le CLCHNAME correspond à ce que vous attendez
2. Consultez le journal des travaux pour vérifier si le processus de commutation est terminé. Si ce n'est pas le cas, attendez et vérifiez la nouvelle file d'attente de transmission du canal ultérieurement.

Si vous utilisez plusieurs files d'attente de transmission de cluster, il est important de concevoir les définitions de files d'attente de transmission de manière explicite et d'éviter les chevauchements de configuration complexes. De cette manière, vous pouvez vous assurer qu'en cas de problèmes, vous pouvez revenir aux files d'attente et à la configuration d'origine.

Si vous rencontrez des problèmes lors du passage à une autre file d'attente de transmission, vous devez résoudre les problèmes avant de procéder à la modification.

Une demande de changement existante doit être terminée pour qu'une nouvelle demande de changement puisse être effectuée. Par exemple, vous:

1. Définissez une nouvelle file d'attente de transmission avec une longueur maximale de 1 et 10 messages sont en attente d'envoi.
2. Modifiez la file d'attente de transmission pour indiquer le nom du canal dans le paramètre CLCHNAME.
3. Arrêtez et redémarrez le canal. La tentative de déplacement des messages échoue et signale les problèmes.
4. Modifiez le paramètre CLCHNAME de la file d'attente de transmission pour qu'il soit à blanc.
5. Arrêtez et redémarrez le canal. Le canal continue d'essayer de terminer la demande d'origine, de sorte qu'il continue d'utiliser la nouvelle file d'attente de transmission.
6. Il est nécessaire de résoudre les problèmes et de redémarrer le canal pour que le déplacement des messages aboutisse.

Lors du prochain redémarrage du canal, il récupère toutes les modifications. Par conséquent, si vous avez mis CLCHNAME à blanc, le canal n'utilisera pas la file d'attente de transmission indiquée.

Dans cet exemple, le fait de remplacer par des blancs le CLCHNAME de la file d'attente de transmission ne signifie pas nécessairement que le canal utilise le système SYSTEM.CLUSTER.TRANSMIT , car il peut y avoir d'autres files d'attente de transmission dont le paramètre CLCHNAME correspond au nom du canal. Par exemple, un nom générique ou l'attribut de gestionnaire de files d'attente DEFCLXQ peut être défini sur channel, de sorte que le canal utilise une file d'attente dynamique à la place de SYSTEM.CLUSTER.TRANSMIT .

Ecriture de programmes pour administrer IBM MQ for z/OS

Vous pouvez écrire vos propres programmes d'application pour administrer un gestionnaire de files d'attente. Utilisez cette rubrique pour comprendre les conditions requises pour l'écriture de vos propres programmes d'administration.

Début des informations d'interface de programmation à usage général

Cet ensemble de rubriques contient des astuces et des conseils pour vous permettre d'émettre des commandes IBM MQ à partir d'un programme d'application IBM MQ .

Remarque : Dans cette rubrique, les appels MQI sont décrits à l'aide de la notation en langage C. Pour les appels classiques des appels dans les langages COBOL, PL/I et assembleur, voir le manuel [Appels de fonction](#) .

Comprendre comment tout fonctionne

En résumé, la procédure d'émission de commandes à partir d'un programme d'application est la suivante:

1. Générez une commande IBM MQ dans un type de message IBM MQ appelé *message de demande*. La commande peut être au format MQSC ou PCF.
2. Envoyer (utilisez MQPUT) ce message dans une file d'attente spéciale appelée file d'entrée des commandes système. Le processeur de commandes IBM MQ exécute la commande.
3. Extraire (utilisez MQGET) les résultats de la commande sous forme de *messages de réponse* dans la file d'attente de réponse. Ces messages contiennent les messages utilisateur dont vous avez besoin pour déterminer si votre commande a abouti et, le cas échéant, quels ont été les résultats.

Ensuite, c'est à votre programme d'application de traiter les résultats.

Cet ensemble de rubriques contient:

Préparation des files d'attente pour les programmes d'administration

Les programmes d'administration nécessitent un certain nombre de files d'attente prédéfinies pour l'entrée de commandes système et la réception de réponses.

Cette section s'applique aux commandes au format MQSC. Pour l'équivalent dans PCF, voir [«Utilisation des formats de commande programmables IBM MQ»](#), à la page 23.

Avant de pouvoir émettre des appels MQPUT ou MQGET , vous devez d'abord définir, puis ouvrir les files d'attente que vous allez utiliser.

Définition de la file d'attente d'entrée des commandes système

La file d'attente d'entrée des commandes système est une file d'attente locale appelée SYSTEM.COMMAND.INPUT. Le fichier d'initialisation CSQINP2 fourni, thlqual.SCSQPROC(CSQ4INSG), contient une définition par défaut pour la file d'attente d'entrée des commandes système. Pour la compatibilité avec IBM MQ sur d'autres plateformes, un alias de cette file d'attente, appelé SYSTEM.ADMIN.COMMAND.QUEUE est également fourni. Pour plus d'informations, voir [Exemples de définitions fournis avec IBM MQ](#) .

Définition d'une file d'attente de réponse

Vous devez définir une file d'attente de réponse pour recevoir les messages de réponse du processeur de commandes IBM MQ . Il peut s'agir de n'importe quelle file d'attente avec des attributs qui permettent d'y placer des messages de réponse. Toutefois, pour un fonctionnement normal, spécifiez les attributs suivants:

- UTILISATION (NORMALE)
- NOTRIGGER (sauf si votre application utilise le déclenchement)

Évitez d'utiliser des messages persistants pour les commandes, mais si vous choisissez de le faire, la file d'attente de réponse ne doit pas être une file d'attente dynamique temporaire.

Le fichier d'initialisation CSQINP2 fourni, thlqual.SCSQPROC(CSQ4INSG), contient une définition pour une file d'attente modèle appelée SYSTEM.COMMAND.REPLY.MODEL. Vous pouvez utiliser ce modèle pour créer une file d'attente de réponse dynamique.

Remarque : La longueur des réponses générées par le processeur de commandes peut atteindre 15 000 octets.

Si vous utilisez une file d'attente dynamique permanente comme file d'attente de réponse, votre application doit laisser le temps à toutes les opérations PUT et GET de se terminer avant de tenter de supprimer la file d'attente, sinon MQRC2055 (MQRC_Q_NOT_EMPTY) peut être renvoyé. Si cela se produit, relancez la suppression de la file d'attente au bout de quelques secondes.

Ouverture de la file d'attente d'entrée des commandes système

Pour que vous puissiez ouvrir la file d'attente d'entrée des commandes système, votre programme d'application doit être connecté à votre gestionnaire de files d'attente. Pour ce faire, utilisez l'appel MQI MQCONN ou MQCONNX .

Utilisez ensuite l'appel MQI MQOPEN pour ouvrir la file d'attente d'entrée des commandes système. Pour utiliser cet appel:

1. Définissez le paramètre **Options** sur MQOO_OUTPUT
2. Définissez les zones de descripteur d'objet MQOD comme suit:

ObjectType

MQOT_Q (l'objet est une file d'attente)

ObjectName

SYSTEM.COMMAND.INPUT

ObjectQMgrName

Si vous souhaitez envoyer vos messages de demande à votre gestionnaire de files d'attente local, laissez cette zone vide. Cela signifie que vos commandes sont traitées localement.

Si vous souhaitez que vos commandes IBM MQ soient traitées sur un gestionnaire de files d'attente éloignées, entrez son nom ici. Vous devez également avoir configuré les files d'attente et les liens appropriés, comme décrit dans [Mise en file d'attente répartie et clusters](#).

Ouverture d'une file d'attente de réponse

Pour extraire les réponses d'une commande IBM MQ , vous devez ouvrir une file d'attente de réponse. L'une des méthodes consiste à spécifier la file d'attente modèle, SYSTEM.COMMAND.REPLY.MODEL dans un appel MQOPEN , pour créer une file d'attente dynamique permanente en tant que file d'attente de réponse. Pour utiliser cet appel:

1. Définissez le paramètre **Options** sur MQOO_INPUT_SHARED
2. Définissez les zones de descripteur d'objet MQOD comme suit:

ObjectType

MQOT_Q (l'objet est une file d'attente)

ObjectName

Nom de la file d'attente de réponse. Si le nom de file d'attente que vous spécifiez est le nom d'un objet file d'attente modèle, le gestionnaire de files d'attente crée une file d'attente dynamique.

ObjectQMgrName

Pour recevoir des réponses sur votre gestionnaire de files d'attente local, laissez cette zone vide.

DynamicQName

Indiquez le nom de la file d'attente dynamique à créer.

Le serveur de commandes est un composant IBM MQ qui fonctionne avec le composant du processeur de commandes. Vous pouvez envoyer des messages formatés au serveur de commandes qui les interprète, exécute les demandes d'administration et renvoie les réponses à votre application d'administration.

Le serveur de commandes lit les messages de demande dans la file d'attente d'entrée des commandes système, les vérifie et les transmet en tant que commandes au processeur de commandes. Le processeur de commandes traite les commandes et place les réponses en tant que messages de réponse dans la file d'attente de réponse que vous spécifiez. Le premier message de réponse contient le message utilisateur CSQN205I. Pour plus d'informations, voir [«Interprétation des messages de réponse du serveur de commandes»](#), à la page 375. Le serveur de commandes traite également les commandes d'initiateur de canal et de groupe de partage de files d'attente, quel que soit l'endroit où elles sont émises.

Identification du gestionnaire de files d'attente qui traite vos commandes

Le gestionnaire de files d'attente qui traite les commandes que vous émettez à partir d'un programme d'administration est le gestionnaire de files d'attente qui possède la file d'attente d'entrée des commandes système dans laquelle le message est inséré.

Démarrage du serveur de commandes

Normalement, le serveur de commandes est démarré automatiquement lorsque le gestionnaire de files d'attente est démarré. Il devient disponible dès que le message CSQ9022I 'START QMGR'NORMAL COMPLETION est renvoyé par la commande START QMGR. Le serveur de commandes est arrêté lorsque toutes les tâches connectées ont été déconnectées lors de la phase d'arrêt du système.

Vous pouvez contrôler le serveur de commandes vous-même à l'aide des commandes START CMDSERV et STOP CMDSERV. Pour empêcher le démarrage automatique du serveur de commandes lors du redémarrage de IBM MQ, vous pouvez ajouter une commande STOP CMDSERV à vos fichiers d'initialisation CSQINP1 ou CSQINP2. Toutefois, cette opération n'est pas recommandée car elle empêche le traitement des commandes d'initiateur de canal ou de groupe de partage de files d'attente.

La commande STOP CMDSERV arrête le serveur de commandes dès qu'il a terminé le traitement du message en cours, ou immédiatement si aucun message n'est en cours de traitement.

Si le serveur de commandes a été arrêté par une commande STOP CMDSERV dans le programme, aucune autre commande du programme ne peut être traitée. Pour redémarrer le serveur de commandes, vous devez exécuter une commande START CMDSERV à partir de la console z/OS.

Si vous arrêtez et redémarrez le serveur de commandes alors que le gestionnaire de files d'attente est en cours d'exécution, tous les messages qui se trouvent dans la file d'attente d'entrée des commandes système lors de l'arrêt du serveur de commandes sont traités lors du redémarrage du serveur de commandes. Toutefois, si vous arrêtez et redémarrez le gestionnaire de files d'attente après l'arrêt du serveur de commandes, seuls les messages persistants de la file d'attente d'entrée de commandes système sont traités lors du redémarrage du serveur de commandes. Tous les messages non persistants de la file d'attente d'entrée des commandes système sont perdus.

Envoi de commandes au serveur de commandes

Pour chaque commande, vous générez un message contenant la commande, puis vous la placez dans la file d'attente d'entrée de la commande système.

Génération d'un message incluant des commandes IBM MQ

Vous pouvez incorporer des commandes IBM MQ dans un programme d'application en créant des messages de demande qui incluent les commandes requises. Pour chaque commande de ce type, vous:

1. Créez une mémoire tampon contenant une chaîne de caractères représentant la commande.
2. Emettez un appel MQPUT en spécifiant le nom de la mémoire tampon dans le paramètre **buffer** de l'appel.

La façon la plus simple de le faire en C est de définir une mémoire tampon à l'aide de 'char'. Exemple :

```
char message_buffer[ ] = "ALTER QLOCAL(SALES) PUT(ENABLED)";
```

Lorsque vous générez une commande, utilisez une chaîne de caractères terminée par une valeur nulle. N'indiquez pas de chaîne de préfixe de commande (CPF) au début d'une commande définie de cette manière. Cela signifie que vous n'avez pas besoin de modifier vos scripts de commande si vous souhaitez les exécuter sur un autre gestionnaire de files d'attente. Toutefois, vous devez tenir compte du fait qu'un CPF est inclus dans les messages de réponse placés dans la file d'attente de réponse.

Le serveur de commandes convertit tous les caractères minuscules en majuscules, sauf s'ils sont placés entre guillemets.

Les commandes peuvent comporter jusqu'à 32 762 caractères.

Insertion de messages dans la file d'attente d'entrée des commandes système

Utilisez l'appel MQPUT pour insérer des messages de demande contenant des commandes dans la file d'attente d'entrée des commandes système. Dans cet appel, vous indiquez le nom de la file d'attente de réponse que vous avez déjà ouverte.

Pour utiliser l'appel MQPUT :

1. Définissez les paramètres MQPUT suivants:

Hconn

Descripteur de connexion renvoyé par l'appel MQCONN ou MQCONNX .

Hobj

Descripteur d'objet renvoyé par l'appel MQOPEN pour la file d'attente d'entrée de la commande système.

BufferLength

Longueur de la commande formatée.

Buffer

Nom de la mémoire tampon contenant la commande.

2. Définissez les zones MQMD suivantes:

MsgType

MQMT_REQUEST

Format

MQFMT_STRING ou MQFMT_NONE

Si vous n'utilisez pas la même page de codes que le gestionnaire de files d'attente, définissez *CodedCharSetId* comme il convient et définissez MQFMT_STRING de sorte que le serveur de commandes puisse convertir le message. Ne définissez pas MQFMT_ADMIN, car cela entraîne l'interprétation de votre commande en tant que PCF.

ReplyToQ

Nom de la file d'attente de réponse.

ReplyToQMGr

Si vous souhaitez que les réponses soient envoyées à votre gestionnaire de files d'attente local, laissez cette zone vide. Si vous souhaitez que vos commandes IBM MQ soient envoyées à un gestionnaire de files d'attente éloignées, entrez son nom ici. Vous devez également avoir configuré les files d'attente et les liens appropriés, comme décrit dans [Mise en file d'attente répartie et clusters](#).

3. Définissez les autres zones MQMD, selon les besoins. Vous devez normalement utiliser des messages non persistants pour les commandes.

4. Définissez les options *PutMsgOpts* , selon les besoins.

Si vous spécifiez MQPMO_SYNCPOINT (valeur par défaut), vous devez suivre l'appel MQPUT avec un appel de point de synchronisation.

Utilisation de MQPUT1 et de la file d'attente d'entrée des commandes système

Si vous souhaitez placer un seul message dans la file d'attente d'entrée de la commande système, vous pouvez utiliser l'appel **MQPUT1** . Cet appel combine les fonctions d'un **MQOPEN**, suivi d'un **MQPUT** d'un message, suivi d'un **MQCLOSE**, le tout en un seul appel. Si vous utilisez cet appel, modifiez les paramètres en conséquence. Pour plus d'informations, voir [Insertion d'un message dans une file d'attente à l'aide de l'appel MQPUT1](#) .

Extraction des réponses à vos commandes

Le serveur de commandes envoie une réponse à une file d'attente de réponses pour chaque message de demande qu'il reçoit. Toute application d'administration doit recevoir et gérer les messages de réponse.

Lorsque le processeur de commandes traite vos commandes, les messages de réponse sont placés dans la file d'attente de réponse spécifiée dans l'appel MQPUT . Le serveur de commandes envoie les messages de réponse avec la même persistance que le message de commande qu'il a reçu.

En attente d'une réponse

Utilisez l'appel MQGET pour extraire une réponse de votre message de demande. Un message de demande peut générer plusieurs messages de réponse. Pour des détails, voir [«Interprétation des messages de réponse du serveur de commandes»](#), à la page 375.

Vous pouvez spécifier un intervalle de temps pendant lequel un appel MQGET attend la génération d'un message de réponse. Si vous n'obtenez pas de réponse, utilisez la liste de contrôle à partir de la rubrique [«Si vous ne recevez pas de réponse»](#), à la page 375.

Pour utiliser l'appel MQGET :

1. Définissez les paramètres suivants:

Hconn

Descripteur de connexion renvoyé par l'appel MQCONN ou MQCONNX .

Hobj

Descripteur d'objet renvoyé par l'appel MQOPEN pour la file d'attente de réponse.

Buffer

Nom de la zone devant recevoir la réponse.

BufferLength

Longueur de la mémoire tampon devant recevoir la réponse. Il doit s'agir d'un minimum de 80 octets.

2. Pour vous assurer que vous obtenez uniquement les réponses de la commande que vous avez émise, vous devez spécifier les zones *MsgId* et *CorrelId* appropriées. Celles-ci dépendent des options de rapport, MQMD_REPORT, que vous avez spécifiées dans l'appel MQPUT :

MQRO_AUCUN

Zéro binaire,'00 ... 00'(24 valeurs nulles).

MQRO_NEW_MSG_ID

Zéro binaire,'00 ... 00'(24 valeurs nulles).

Il s'agit de la valeur par défaut si aucune de ces options n'a été spécifiée.

MQRO_PASS_MSG_ID

Le *MsgId* de la **MQPUT**.

MQRO_AUCUN

Le *MsgId* de l'appel MQPUT .

MQRO_COPY_MSG_ID_TO_CORREL_ID

Le *MsgId* de l'appel MQPUT .

Il s'agit de la valeur par défaut si aucune de ces options n'a été spécifiée.

MQRO_PASS_CORREL_ID

Le *CorrelId* de l'appel MQPUT .

Pour plus de détails sur les options de rapport, voir [Options de rapport et indicateurs de message](#).

3. Définissez les zones *GetMsgOpts* suivantes:**Options**

MQGMO_WAIT

Si vous n'utilisez pas la même page de codes que le gestionnaire de files d'attente, définissez MQGMO_CONVERT et définissez *CodedCharSetId* comme approprié dans le MQMD.

WaitInterval

Pour les réponses du gestionnaire de files d'attente local, essayez 5 secondes. Codée en millisecondes, elle devient 5 000. Pour les réponses provenant d'un gestionnaire de files d'attente éloignées et les commandes de contrôle de canal et de statut, essayez 30 secondes. Codée en millisecondes, elle devient 30 000.

Messages supprimés

Si le serveur de commandes détecte qu'un message de demande n'est pas valide, il supprime ce message et écrit le message CSQN205I dans la file d'attente de réponse nommée. S'il n'existe pas de file d'attente de réponse, le message CSQN205I est inséré dans la file d'attente de rebut. Le code retour de ce message indique la raison pour laquelle le message de demande d'origine n'était pas valide:

- 00D5020F** Il n'est pas de type MQMT_REQUEST.
- 00D50210** Sa longueur est nulle.
- 00D50212** Sa longueur est supérieure à 32 762 octets.
- 00D50211** Il contient tous les blancs.
- 00D5483E** La conversion était nécessaire, mais *Format* n'était pas MQFMT_STRING.
- Autre** Voir [Codes de serveur de commandes](#)

Descripteur de message de réponse du serveur de commandes

Pour tout message de réponse, les zones de descripteur de message MQMD suivantes sont définies:

| | |
|--------------------|--|
| <i>MsgType</i> | MQMT_REPLY |
| <i>Feedback</i> | MQFB_AUCUN |
| <i>Encoding</i> | MQENC_NATIVE |
| <i>Priority</i> | Comme pour le MQMD dans le message que vous avez émis. |
| <i>Persistence</i> | Comme pour le MQMD dans le message que vous avez émis. |
| <i>CorrelId</i> | Dépend des options de rapport MQPUT . |
| <i>ReplyToQ</i> | Aucune. |

Le serveur de commandes définit la zone *Options* de la structure MQPMO sur MQPMO_NO_SYNCPOINT. Cela signifie que vous pouvez extraire les réponses au fur et à mesure de leur création, plutôt qu'en tant que groupe au point de synchronisation suivant.

Interprétation des messages de réponse du serveur de commandes

Chaque message de demande correctement traité par IBM MQ génère au moins deux messages de réponse. Chaque message de réponse contient un seul message utilisateur IBM MQ .

La longueur d'une réponse dépend de la commande qui a été émise. La réponse la plus longue que vous pouvez obtenir provient d'une commande DISPLAY NAMELIST, qui peut comporter jusqu'à 15 000 octets.

Le premier message utilisateur, CSQN205I, contient toujours:

- Nombre de réponses (en décimal), que vous pouvez utiliser comme compteur dans une boucle pour obtenir le reste des réponses. Le comptage inclut ce premier message.
- Code retour du préprocesseur de commandes.
- Code anomalie, qui correspond au code anomalie du processeur de commandes.

Ce message ne contient pas de CPF.

Exemple :

```
CSQN205I    COUNT=    4, RETURN=0000000C, REASON=00000008
```

La zone COUNT a une longueur de 8 octets et est justifiée à droite. Il commence toujours à la position 18, c'est-à-dire immédiatement après COUNT=. La zone RETURN a une longueur de 8 octets en caractères hexadécimaux et se trouve immédiatement après RETURN= à la position 35. La zone REASON a une longueur de 8 octets en caractères hexadécimaux et se trouve immédiatement après REASON= à la position 52.

Si la valeur RETURN= est 00000000 et que la valeur REASON= est 00000004, l'ensemble des messages de réponse est incomplet. Après avoir extrait les réponses indiquées par le message CSQN205I , émettez un autre appel MQGET pour attendre un autre ensemble de réponses. Le premier message de l'ensemble de réponses suivant est à nouveau CSQN205I, indiquant le nombre de réponses et s'il reste encore à venir.

Pour plus de détails sur les messages individuels, voir la documentation [Messages, codes achèvement et codes anomalie IBM MQ for z/OS](#) .

Si vous utilisez une fonction autre que l'anglais, le texte et la présentation des réponses sont différents de ceux affichés ici. Toutefois, la taille et la position du nombre et des codes retour du message CSQN205I sont identiques.

Si vous ne recevez pas de réponse

Vous pouvez effectuer une série d'étapes si vous ne recevez pas de réponse à une demande adressée au serveur de commandes.

Si vous ne recevez pas de réponse à votre message de demande, consultez la liste de contrôle suivante:

- Le serveur de commandes est-il en cours d'exécution?
- Le *WaitInterval* est-il assez long?
- Les files d'attente d'entrée et de réponse de la commande système sont-elles correctement définies?
- Les appels MQOPEN à ces files d'attente ont-ils abouti?
- Les files d'attente d'entrée de commande système et de réponse sont-elles activées pour les appels MQPUT et MQGET ?

- Avez-vous envisagé d'augmenter les attributs MAXDEPTH et MAXMSGL de vos files d'attente?
- Utilisez-vous correctement les zones *CorrelId* et *MsgId* ?
- Le gestionnaire de files d'attente est-il toujours en cours d'exécution?
- La commande a-t-elle été générée correctement?
- Tous vos liens distants sont-ils définis et fonctionnent correctement?
- Les appels MQPUT ont-ils été correctement définis?
- La file d'attente de réponse a-t-elle été définie en tant que file d'attente dynamique temporaire au lieu d'une file d'attente dynamique permanente? (Si le message de demande est persistant, vous devez utiliser une file d'attente dynamique permanente pour la réponse.)

Lorsque le serveur de commandes génère des réponses mais ne peut pas les écrire dans la file d'attente de réponse que vous spécifiez, il les écrit dans la file d'attente de rebut.

z/OS Transmission de commandes à l'aide de MGCRE

Avec les autorisations appropriées, un programme d'application peut envoyer des demandes à plusieurs gestionnaires de files d'attente à l'aide d'une routine de service z/OS .

Si vous disposez de l'autorisation appropriée, vous pouvez transmettre des commandes IBM MQ de votre programme à plusieurs gestionnaires de files d'attente par le service z/OS MGCRE (SVC 34). La valeur de CPF identifie le gestionnaire de files d'attente particulier vers lequel la commande est dirigée. Pour plus d'informations sur les CPFs, voir [User IDs for command security and command resource security](#) et «Emission de commandes de gestionnaire de files d'attente sous z/OS», à la page 347.

Si vous utilisez MGCRE, vous pouvez utiliser un jeton de commande et de réponse (CART) pour obtenir les réponses directes à la commande.

z/OS Exemples de commandes et leurs réponses

Utilisez cette rubrique comme une série d'exemples de commandes destinées au serveur de commandes et aux réponses du serveur de commandes.

Voici quelques exemples de commandes pouvant être intégrées aux messages IBM MQ et aux messages utilisateur correspondant aux réponses. Sauf indication contraire, chaque ligne de la réponse est un message distinct.

- [Messages d'une commande DEFINE](#)
- [Messages d'une commande DELETE](#)
- [Messages des commandes DISPLAY](#)
- [Messages des commandes avec CMDSCOPE](#)
- [Messages des commandes qui génèrent des commandes avec CMDSCOPE](#)

Messages d'une commande DEFINE

the following command:

```
DEFINE QLOCAL(Q1)
```

génère les messages suivants:

```
CSQN205I    COUNT=    2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQMMSGP ' DEFINE QLOCAL' NORMAL COMPLETION
```


Ces messages de réponse sont générés à la fin normale.

Messages d'une commande DELETE

the following command:

```
DELETE QLOCAL(Q2)
```

génère les messages suivants:

```
CSQN205I    COUNT=    4, RETURN=0000000C, REASON=00000008  
CSQM125I +CSQ1 CSQMUQLC QLOCAL (Q2) QSGDISP(QMGR) WAS NOT FOUND  
CSQM090E +CSQ1 CSQMUQLC FAILURE REASON CODE X'00D44002'  
CSQ9023E +CSQ1 CSQMUQLC ' DELETE QLOCAL' ABNORMAL COMPLETION
```

Ces messages indiquent qu'une file d'attente locale appelée Q2 n'existe pas.

Messages des commandes DISPLAY

Les exemples suivants montrent les réponses de certaines commandes DISPLAY.

Recherche du nom de la file d'attente de rebut

Si vous souhaitez connaître le nom de la file d'attente de rebut d'un gestionnaire de files d'attente, exécutez la commande suivante à partir d'un programme d'application:

```
DISPLAY QMGR DEADQ
```

Les trois messages utilisateur suivants sont renvoyés, à partir desquels vous pouvez extraire le nom requis:

```
CSQN205I    COUNT=    3, RETURN=00000000, REASON=00000000  
CSQM409I +CSQ1 QMNAME(CSQ1) DEADQ(SYSTEM.DEAD.QUEUE  
CSQ9022I +CSQ1 CSQMDRTS ' DISPLAY QMGR' NORMAL COMPLETION
```

Messages de la commande DISPLAY QUEUE

Les exemples suivants montrent comment les résultats d'une commande dépendent des attributs spécifiés dans cette commande.

Exemple 1

Vous définissez une file d'attente locale à l'aide de la commande suivante:

```
DEFINE QLOCAL(Q1) DESCR('A sample queue') GET(ENABLED) SHARE
```

Si vous exécutez la commande suivante à partir d'un programme d'application:

```
DISPLAY QUEUE(Q1) SHARE GET DESCR
```

Les trois messages utilisateur suivants sont renvoyés:

```
CSQN205I  COUNT=    3, RETURN=00000000, REASON=00000000
CSQM401I +CSQ1 QUEUE(Q1                                ) TYPE(
QLOCAL ) QSGDISP(QMGR  )
DESCR(A sample queue
) SHARE GET(ENABLED )
CSQ9022I +CSQ1 CSQMDMSG ' DISPLAY QUEUE' NORMAL COMPLETION
```

Remarque : Le deuxième message, CSQM401I, apparaît ici sur quatre lignes.

Exemple 2

Deux files d'attente ont des noms commençant par la lettre A:

- A1 est une file d'attente locale dont l'attribut PUT est défini sur DISABLED.
- A2 est une file d'attente éloignée dont l'attribut PUT est défini sur ENABLED.

Si vous exécutez la commande suivante à partir d'un programme d'application:

```
DISPLAY QUEUE(A*) PUT
```

Les quatre messages utilisateur suivants sont renvoyés:

```
CSQN205I  COUNT=    4, RETURN=00000000, REASON=00000000
CSQM401I +CSQ1 QUEUE(A1                                ) TYPE(
QLOCAL ) QSGDISP(QMGR  )
PUT(DISABLED )
CSQM406I +CSQ1 QUEUE(A2                                ) TYPE(
QREMOTE ) PUT(ENABLED )
CSQ9022I +CSQ1 CSQMDMSG ' DISPLAY QUEUE' NORMAL COMPLETION
```

Remarque : Les deuxième et troisième messages, CSQM401I et CSQM406I, sont affichés ici sur trois et deux lignes.

Messages de la commande DISPLAY NAMELIST

Vous définissez une liste de noms à l'aide de la commande suivante:

```
DEFINE NAMELIST(N1) NAMES(Q1,SAMPLE_QUEUE)
```

Si vous exécutez la commande suivante à partir d'un programme d'application:

```
DISPLAY NAMELIST(N1) NAMES NAMCOUNT
```

les trois messages utilisateur suivants sont renvoyés:

```
CSQN205I  COUNT=    3, RETURN=00000000, REASON=00000000
CSQM407I +CSQ1 NAMELIST(N1                             ) QS
GDISP(QMGR  ) NAMCOUNT(    2) NAMES(Q1
,SAMPLE_QUEUE
)
CSQ9022I +CSQ1 CSQMDMSG ' DISPLAY NAMELIST' NORMAL COMPLETION
```

Remarque : Le deuxième message, CSQM407I, apparaît ici sur trois lignes.

Messages des commandes avec CMDSCOPE

Les exemples suivants montrent les réponses des commandes qui ont été entrées avec l'attribut CMDSCOPE.

Messages de la commande ALTER PROCESS

the following command:

```
ALT PRO(V4) CMDSCOPE(*)
```

génère les messages suivants:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'ALT PRO' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 5, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'ALT PRO' command responses from MQ26
CSQM125I !MQ26 CSQMMSGP PROCESS(V4) QSGDISP(QMGR) WAS NOT FOUND
CSQM090E !MQ26 CSQMMSGP FAILURE REASON CODE X'00D44002'
CSQ9023E !MQ26 CSQMMSGP 'ALT PRO' ABNORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'ALT PRO' command responses from MQ25
CSQ9022I !MQ25 CSQMMSGP 'ALT PRO' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=0000000C, REASON=00000008
CSQN123E !MQ25 'ALT PRO' command for CMDSCOPE(*) abnormal completion
```

Ces messages vous indiquent que la commande a été entrée sur le gestionnaire de files d'attente MQ25 et envoyée à deux gestionnaires de files d'attente (MQ25 et MQ26). La commande a abouti sur MQ25 mais la définition de processus n'existait pas sur MQ26. Par conséquent, la commande a échoué sur ce gestionnaire de files d'attente.

Messages de la commande DISPLAY PROCESS

the following command:

```
DIS PRO(V*) CMDSCOPE(*)
```

génère les messages suivants:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'DIS PRO' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 5, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS PRO' command responses from MQ26
CSQM408I !MQ26 PROCESS(V2) QSGDISP(COPY)
CSQM408I !MQ26 PROCESS(V3) QSGDISP(QMGR)
CSQ9022I !MQ26 CSQMDRTS 'DIS PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 7, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS PRO' command responses from MQ25
CSQM408I !MQ25 PROCESS(V2) QSGDISP(COPY)
CSQM408I !MQ25 PROCESS(V2) QSGDISP(GROUP)
CSQM408I !MQ25 PROCESS(V3) QSGDISP(QMGR)
CSQM408I !MQ25 PROCESS(V4) QSGDISP(QMGR)
CSQ9022I !MQ25 CSQMDRTS 'DIS PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DIS PRO' command for CMDSCOPE(*) normal completion
```

Ces messages vous indiquent que la commande a été entrée sur le gestionnaire de files d'attente MQ25 et envoyée à deux gestionnaires de files d'attente (MQ25 et MQ26). Des informations sont

affichées sur tous les processus de chaque gestionnaire de files d'attente dont le nom commence par la lettre V.

Messages de la commande DISPLAY CHSTATUS

the following command:

```
DIS CHS(VT) CMDSCOPE(*)
```

génère les messages suivants:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'DIS CHS' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 4, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS CHS' command responses from MQ25
CSQM422I !MQ25 CHSTATUS(VT) CHLDISP(PRIVATE) CONNAME( ) CURRENT STATUS(STOPPED)
CSQ9022I !MQ25 CSQXDRTS ' DIS CHS' NORMAL COMPLETION
CSQN205I COUNT= 4, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DIS CHS' command responses from MQ26
CSQM422I !MQ26 CHSTATUS(VT) CHLDISP(PRIVATE) CONNAME( ) CURRENT STATUS(STOPPED)
CSQ9022I !MQ26 CSQXDRTS ' DIS CHS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DIS CHS' command for CMDSCOPE(*) normal completion
```

Ces messages vous indiquent que la commande a été entrée sur le gestionnaire de files d'attente MQ25 et envoyée à deux gestionnaires de files d'attente (MQ25 et MQ26). Des informations sont affichées sur le statut des canaux de chaque gestionnaire de files d'attente.

Messages de la commande STOP CHANNEL

the following command:

```
STOP CHL(VT) CMDSCOPE(*)
```

génère les messages suivants:

```
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQN137I !MQ25 'STOP CHL' command accepted for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ25
CSQM134I !MQ25 CSQMTCHL STOP CHL(VT) COMMAND ACCEPTED
SQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ26
CSQM134I !MQ26 CSQMTCHL STOP CHL(VT) COMMAND ACCEPTED
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ26
CSQ9022I !MQ26 CSQXCRPS ' STOP CHL' NORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'STOP CHL' command responses from MQ25
CSQ9022I !MQ25 CSQXCRPS ' STOP CHL' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'STOP CHL' command for CMDSCOPE(*) normal completion
```

Ces messages vous indiquent que la commande a été entrée sur le gestionnaire de files d'attente MQ25 et envoyée à deux gestionnaires de files d'attente (MQ25 et MQ26). Le canal VT a été arrêté sur chaque gestionnaire de files d'attente.

Messages des commandes qui génèrent des commandes avec CMDSCOPE

the following command:

```
DEF PRO(V2) QSGDISP(GROUP)
```

génère les messages suivants:

```
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQM122I !MQ25 CSQMMSGP ' DEF PRO' COMPLETED FOR QSGDISP(GROUP)
CSQN138I !MQ25 'DEFINE PRO' command generated for CMDSCOPE(*), sent to 2
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DEFINE PRO' command responses from MQ25
CSQ9022I !MQ25 CSQMMSGP ' DEFINE PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 3, RETURN=00000000, REASON=00000004
CSQN121I !MQ25 'DEFINE PRO' command responses from MQ26
CSQ9022I !MQ26 CSQMMSGP ' DEFINE PROCESS' NORMAL COMPLETION
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQN122I !MQ25 'DEFINE PRO' command for CMDSCOPE(*) normal completion
```

Ces messages vous indiquent que la commande a été entrée sur le gestionnaire de files d'attente MQ25. Lorsque l'objet a été créé dans le référentiel partagé, une autre commande a été générée et envoyée à tous les gestionnaires de files d'attente actifs du groupe de partage de files d'attente (MQ25 et MQ26).

z/OS

Gestion des ressources IBM MQ sur z/OS

Utilisez les liens de cette rubrique pour savoir comment gérer les ressources utilisées par IBM MQ for z/OS, par exemple, la gestion des fichiers journaux, des ensembles de données, des ensembles de pages, des pools de mémoire tampon et des structures d'unité de couplage.

Utilisez les liens suivants pour obtenir des détails sur les différentes tâches d'administration que vous devrez peut-être effectuer lors de l'utilisation de IBM MQ for z/OS:

- [«Gestion des journaux», à la page 382](#)
- [«Gestion du fichier d'amorçage \(BSDS\)», à la page 391](#)
- [«Gestion des ensembles de pages», à la page 399](#)
- [«Comment sauvegarder et récupérer des ensembles de pages», à la page 406](#)
- [«Comment sauvegarder et restaurer des files d'attente à l'aide de CSQUTIL», à la page 411](#)
- [«Gestion des pools de mémoire tampon», à la page 411](#)
- [«Gestion des groupes de partage de files d'attente et des files d'attente partagées sous z/OS», à la page 412](#)

Concepts associés

[«Administration d'IBM MQ for z/OS», à la page 334](#)

L'administration des gestionnaires de files d'attente et des ressources associées inclut les tâches que vous effectuez fréquemment pour activer et gérer ces ressources. Choisissez la méthode que vous préférez pour administrer vos gestionnaires de files d'attente et les ressources associées.

[«Emission de commandes sur IBM MQ for z/OS», à la page 335](#)

Vous pouvez utiliser des commandes de script IBM MQ (MQSC) en mode de traitement par lots ou en mode interactif pour contrôler un gestionnaire de files d'attente.

[«Reprise et redémarrage sur z/OS», à la page 423](#)

Utilisez cette rubrique pour comprendre les mécanismes de reprise et de redémarrage utilisés par IBM MQ.

Référence associée

[«Les utilitaires IBM MQ for z/OS», à la page 344](#)

IBM MQ for z/OS fournit un ensemble de programmes utilitaires que vous pouvez utiliser pour faciliter l'administration du système.

Information associée

[Concepts d'IBM MQ for z/OS](#)

[Planification de votre environnement IBM MQ sur z/OS](#)

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

[Références relatives aux formats de commandes programmables](#)

[Référence MQSC](#)

[Utilisation des utilitaires IBM MQ for z/OS](#)

Gestion des journaux

Cette rubrique explique comment gérer vos fichiers journaux IBM MQ , y compris le processus d'archivage des journaux, à l'aide de la compression des enregistrements de journal, de la récupération des enregistrements de journal et de l'impression des enregistrements de journal.

Cette rubrique décrit les tâches impliquées dans la gestion des journaux IBM MQ . Il contient les sections suivantes:

Archivage des journaux à l'aide de la commande ARCHIVE LOG

Un opérateur autorisé peut archiver les fichiers journaux actifs IBM MQ en cours chaque fois que cela est nécessaire à l'aide de la commande ARCHIVE LOG.

Lorsque vous exécutez la commande ARCHIVE LOG , IBM MQ tronque les fichiers journaux actifs en cours, puis exécute un processus de déchargement asynchrone et met à jour le fichier d'amorce avec un enregistrement du processus de déchargement.

La commande ARCHIVE LOG comporte une option MODE (QUIESCE). Avec cette option, les travaux et les utilisateurs IBM MQ sont mis au repos après un point de validation et le point de cohérence résultant est capturé dans le journal actif en cours avant d'être déchargé.

Envisagez d'utiliser l'option MODE (QUIESCE) lors de la planification d'une stratégie de sauvegarde pour la reprise hors site. Il crée un point de cohérence à l'échelle du système, ce qui réduit le nombre d'incohérences de données lorsque le journal d'archivage est utilisé avec la copie de groupe de pages de sauvegarde la plus récente lors de la récupération. Exemple :

```
ARCHIVE LOG MODE(QUIESCE)
```

Si vous exécutez la commande ARCHIVE LOG sans spécifier de paramètre TIME, la période de mise au repos prend par défaut la valeur du paramètre QUIESCE de la macro CSQ6ARVP . Si le temps nécessaire à l'exécution du mode ARCHIVE LOG MODE (QUIESCE) est inférieur au temps spécifié, la commande aboutit ; sinon, la commande échoue à l'expiration du délai. Vous pouvez spécifier la période de temps de manière explicite à l'aide de l'option TIME, par exemple:

```
ARCHIVE LOG MODE(QUIESCE) TIME(60)
```

Cette commande indique une période de mise au repos pouvant aller jusqu'à 60 secondes avant l'exécution du traitement ARCHIVE LOG.

Avvertissement : L'utilisation de l'option TIME lorsque le temps est critique peut perturber de manière significative la disponibilité de IBM MQ pour tous les travaux et utilisateurs qui utilisent des ressources IBM MQ .

Par défaut, la commande est traitée de manière asynchrone à partir du moment où vous la soumettez. (Pour traiter la commande de manière synchrone avec d'autres commandes IBM MQ , utilisez l'option

WAIT (YES) avec QUIESCE, mais sachez que la console z/OS est verrouillée à partir de l'entrée de la commande IBM MQ pendant toute la période QUIESCE.)

Pendant la période de mise au repos:

- Les travaux et les utilisateurs du gestionnaire de files d'attente sont autorisés à passer par le processus de validation, mais ils sont suspendus s'ils tentent de mettre à jour une ressource IBM MQ après la validation.
- Les travaux et les utilisateurs qui ne lisent que des données peuvent être affectés, car ils peuvent être en attente de verrous détenus par des travaux ou des utilisateurs qui ont été suspendus.
- De nouvelles tâches peuvent démarrer, mais elles ne peuvent pas mettre à jour les données.

La sortie de la commande DISPLAY LOG utilise le message CSQV400I pour indiquer qu'une mise au repos est activée. Exemple :

```
CSQJ322I +CSQ1 DISPLAY LOG report ...
Parameter  Initial value      SET value
-----
INBUFF     60
OUTBUFF    400
MAXRTU     2
MAXARCH    2
TWOACTV    YES
TWOARCH    YES
TWOBSDS    YES
OFFLOAD    YES
MAXCNOFF   0
WRTHRSH    20
DEALLCT    0
COMPLOG    NONE
ZHYWRITE   NO
End of LOG report
CSQJ370I +CSQ1 LOG status report ...
Copy %Full PPRC DSName
  1      68 NO  VICY.CSQ1.LOGCOPY1.DS01
  2      68 NO  VICY.CSQ1.LOGCOPY2.DS01
Restarted at 2014-04-15 09:49:30 using RBA=000000000891B000
Latest RBA=000000000891CCF8
Offload task is AVAILABLE
Full logs to offload - 0 of 4
CSQV400I +CSQ1 ARCHIVE LOG QUIESCE CURRENTLY ACTIVE
CSQ9022I +CSQ1 CSQJC001 ' DISPLAY LOG ' NORMAL COMPLETION
```



Avertissement : zHyperL'écriture n'étant pas activée dans IBM MQ 9.0 , *NO* est la seule valeur autorisée.

Lorsque toutes les mises à jour sont mises au repos, l'enregistrement de l'historique de mise au repos dans le fichier d'amorce est mis à jour avec la date et l'heure auxquelles les fichiers journaux actifs ont été tronqués et avec la dernière adresse relative en octets écrite dans les fichiers journaux actifs en cours. IBM MQ tronque les fichiers journaux actifs en cours, passe aux fichiers journaux actifs disponibles suivants et émet le message CSQJ311I indiquant que le processus de déchargement a démarré.

Si les mises à jour ne peuvent pas être mises au repos avant l'expiration de la période de mise au repos, IBM MQ émet le message CSQJ317I et le traitement du journal d'archivage s'arrête. Les fichiers journaux actifs en cours ne sont pas tronqués, ni basculés vers les fichiers journaux disponibles suivants, et le processus de déchargement n'est pas démarré.

Que la mise au repos ait abouti ou non, tous les utilisateurs et travaux suspendus sont repris, et IBM MQ émet le message CSQJ312I, indiquant que la mise au repos est terminée et que l'activité de mise à jour est reprise.

Si ARCHIVE LOG est émis alors que le journal actif en cours est le dernier fichier journal actif disponible, la commande n'est pas traitée et IBM MQ émet le message suivant:

```
CSQJ319I - csect-name CURRENT ACTIVE LOG DATA SET IS THE LAST
AVAILABLE ACTIVE LOG DATA SET. ARCHIVE LOG PROCESSING
WILL BE TERMINATED
```

Si ARCHIVE LOG est émis alors qu'une autre commande ARCHIVE LOG est déjà en cours, la nouvelle commande n'est pas traitée et IBM MQ émet le message suivant:

```
CSQJ318I - ARCHIVE LOG COMMAND ALREADY IN PROGRESS
```

Pour plus d'informations sur les messages émis lors de l'archivage, voir [Messages for IBM MQ for z/OS](#).

Redémarrage du processus d'archivage des journaux après un échec

En cas de problème lors du processus d'archivage des journaux (par exemple, un problème d'allocation ou de montage de bande), l'archivage du journal actif peut être suspendu. Vous pouvez annuler le processus d'archivage et le redémarrer à l'aide de la commande ARCHIVE LOG CANCEL OFFLOAD. Cette commande annule tout traitement de déchargement en cours et redémarre le processus d'archivage. Il commence par le fichier journal le plus ancien qui n'a pas été archivé et parcourt tous les fichiers journaux actifs qui doivent être déchargés. Toutes les opérations d'archivage des journaux qui ont été interrompues sont redémarrées.

Utilisez cette commande uniquement si vous êtes certain que la tâche d'archivage des journaux en cours ne fonctionne plus ou si vous souhaitez redémarrer une tentative précédente qui a échoué. En effet, la commande peut entraîner un arrêt anormal de la tâche de déchargement, ce qui peut entraîner un vidage.

Contrôle de l'archivage et de la journalisation

Vous pouvez contrôler la compression, l'impression, l'archivage, la récupération et la journalisation à l'aide des macros CSQ6LOGP, CSQ6ARVP et CSQ6SYSP. Notez que les modifications apportées aux objets privés uniquement sont consignées dans les journaux IBM MQ. Les modifications apportées aux objets GROUP (comme les canaux entrants partagés) sont également consignées, car les définitions sont propagées autour du groupe et conservées en local.

De nombreux aspects de l'archivage et de la journalisation sont contrôlés par des paramètres définis à l'aide des macros CSQ6LOGP, CSQ6ARVP et CSQ6SYSP du module des paramètres système lorsque le gestionnaire de files d'attente est personnalisé. Pour plus d'informations sur ces macros, voir [Tâche 17: Personnalisation de votre module de paramètres système](#).

Certains de ces paramètres peuvent être modifiés pendant l'exécution d'un gestionnaire de files d'attente à l'aide des commandes IBM MQ MQSC SET LOG, SET SYSTEM et SET ARCHIVE. Ils sont présentés dans la [Tableau 26](#), à la page 384:

| <i>Tableau 26. Paramètres d'archivage et de consignation pouvant être modifiés lorsqu'un gestionnaire de files d'attente est en cours d'exécution</i> | |
|---|--|
| commande set | Paramètres |
| Journal | WRTHRSH, MAXARCH, DEALLCT, MAXRTU, COMPLOG |
| ARCHIVE | Tous |
| SYSTEME | LOGLOAD |

Vous pouvez afficher les paramètres de tous les paramètres à l'aide des commandes MQSC [DISPLAY LOG](#), [DISPLAY ARCHIVE](#) et [DISPLAY SYSTEM](#) . Ces commandes affichent également des informations de statut sur l'archivage et la journalisation.

contrôle de la compression de journal

Vous pouvez activer et désactiver la compression des enregistrements de journal à l'aide de l'une des options suivantes:

- Les commandes SET et DISPLAY LOG dans MQSC ; voir [Les commandes MQSC](#)
- Appel de l'interface PCF. Voir «[Introduction à IBM MQ Programmable Command Formats](#)», à la page 22
- Utilisation de la macro CSQ6LOGP dans le module des paramètres système ; voir [Utilisation de CSQ6LOGP](#)

Impression des enregistrements de journal

Vous pouvez extraire et imprimer des enregistrements de journal à l'aide de l'utilitaire CSQ1LOGP . Pour obtenir des instructions, voir [L'utilitaire d'impression de journal](#).

Récupération des journaux

En règle générale, vous n'avez pas besoin de sauvegarder et de restaurer les journaux IBM MQ , en particulier si vous utilisez la double consignation. Toutefois, dans de rares cas, tels qu'une erreur d'E-S dans un journal, vous devrez peut-être récupérer les journaux. Utilisez les services de méthode d'accès pour supprimer et redéfinir le fichier, puis copiez le double journal correspondant dans le fichier.

Suppression des fichiers journaux archivés

Vous pouvez supprimer vos fichiers journaux d'archivage et choisir de les supprimer automatiquement ou manuellement.

Vous devez conserver suffisamment de données de journal pour pouvoir effectuer une reprise sur unité de travail, une reprise sur support d'ensemble de pages en cas de perte d'un ensemble de pages, ou une reprise sur support de structure d'unité de couplage en cas de perte d'une structure d'unité de couplage. Ne supprimez pas les fichiers journaux d'archivage qui peuvent être requis pour la récupération ; si vous supprimez ces fichiers journaux d'archivage, vous risquez de ne pas être en mesure d'effectuer les opérations de récupération requises.

Si vous avez confirmé que vos fichiers journaux archivés peuvent être supprimés, vous pouvez procéder de l'une des manières suivantes:

- [Suppression automatique des fichiers journaux d'archivage](#)
- [Suppression manuelle des fichiers journaux archivés](#)

Suppression automatique des fichiers journaux d'archivage

Vous pouvez utiliser une unité de stockage à accès direct ou un système de gestion de bande pour supprimer automatiquement les fichiers journaux archivés. La durée de conservation des fichiers d'archivage IBM MQ est spécifiée par la zone de durée de conservation ARCRETN dans la macro d'installation CSQ6ARVP (pour plus d'informations, voir [Utilisation de CSQ6ARVP](#)).

La valeur par défaut de la période de conservation indique que les journaux d'archivage doivent être conservés pendant 9999 jours (maximum).

Important : Vous pouvez modifier la durée de conservation, mais vous devez vous assurer que vous pouvez prendre en charge le nombre de cycles de sauvegarde que vous avez planifiés.

IBM MQ utilise la valeur de durée de conservation comme valeur du paramètre JCL RETPD lorsque des fichiers journaux d'archivage sont créés.

La durée de conservation définie par le sous-système de gestion de stockage MVS/DFP (SMS) peut être remplacée par ce paramètre IBM MQ . En règle générale, la durée de conservation est définie sur la valeur la plus faible spécifiée par IBM MQ ou SMS. L'administrateur de stockage et l'administrateur IBM MQ doivent convenir d'une valeur de durée de conservation appropriée pour IBM MQ.

Remarque : IBM MQ ne dispose pas d'une méthode automatisée pour supprimer des informations sur les fichiers d'archivage du fichier d'amorce, car certains systèmes de gestion de bande fournissent des remplacements manuels externes des durées de conservation. Par conséquent, les informations relatives à un fichier d'archivage peuvent toujours se trouver dans le fichier d'amorce longtemps après l'expiration de la période de conservation du fichier et le fichier a été supprimé par le système de gestion de bande. A l'inverse, le nombre maximal de fichiers journaux d'archivage peut avoir été dépassé et les données du fichier d'amorce peuvent avoir été supprimées avant que le fichier n'ait atteint sa date d'expiration.

Si les fichiers journaux d'archivage sont supprimés automatiquement, n'oubliez pas que l'opération ne met pas à jour la liste des journaux d'archivage dans le fichier d'amorce. Vous pouvez mettre à jour le fichier d'amorce à l'aide de l'utilitaire de modification de l'inventaire des journaux, comme décrit dans «[Modification du fichier d'amorce](#)», à la page 392. La mise à jour n'est pas essentielle. L'enregistrement des anciens journaux d'archivage gaspille de l'espace dans le fichier d'amorce, mais ne fait pas d'autres dégâts.

Suppression manuelle des fichiers journaux archivés

Vous devez conserver tous les enregistrements de journal jusqu'à l'adresse relative en octets la plus basse identifiée dans les messages CSQI024I et CSQI025I. Cette adresse relative en octets est obtenue à l'aide de la commande DISPLAY USAGE que vous avez émise lors de la création d'un point de récupération à l'aide de la [méthode 1: sauvegarde intégrale](#).

Lisez la rubrique [Création d'un point de récupération pour les ressources non partagées avant de supprimer des journaux](#).

Localisation et suppression des fichiers journaux d'archivage

Après avoir établi l'adresse relative en octets de journal minimale requise pour la récupération, vous pouvez trouver des fichiers d'archivage contenant uniquement des enregistrements de journal antérieurs en procédant comme suit:

1. Utilisez l'utilitaire d'impression de mappe de journal pour imprimer le contenu du fichier d'amorce. Pour un exemple de sortie, voir [The print log map utility](#).
2. Recherchez les sections de la sortie intitulée ARCHIVE LOG COPY n DATA SETS. Si vous utilisez la double consignation, il y a deux sections. Les colonnes intitulées STARTRBA et ENDRBA indiquent la plage d'adresses relatives en octets contenue dans chaque volume. Recherchez les volumes dont les plages incluent l'adresse RBA minimale que vous avez trouvée avec les messages CSQI024I et CSQI025I. Il s'agit des premiers volumes que vous devez conserver. Si vous utilisez la double consignation, il existe deux volumes de ce type.

Si aucun volume n'a une plage appropriée, l'un des cas suivants s'applique:

- L'adresse relative en octets minimale n'a pas encore été archivée et vous pouvez supprimer tous les volumes du journal d'archivage.
- La liste des volumes de journaux d'archivage dans le fichier d'amorce est encapsulée lorsque le nombre de volumes dépasse le nombre autorisé par le paramètre MAXARCH de la macro CSQ6LOGP . Si le fichier d'amorce n'enregistre pas de volume de journal d'archivage, ce volume ne peut pas être utilisé pour la reprise. Par conséquent, envisagez d'ajouter des informations sur les volumes existants dans le fichier d'amorce. Pour des instructions à ce sujet, voir «[Modifications apportées aux journaux d'archivage](#)», à la page 395.

Envisagez également d'augmenter la valeur de MAXARCH. Pour plus d'informations, voir [Utilisation de CSQ6LOGP](#).

3. Supprimez tout fichier journal d'archivage ou volume dont la valeur ENDRBA est inférieure à la valeur STARTRBA du volume le plus ancien que vous souhaitez conserver. Si vous utilisez la double consignation, supprimez ces deux copies.

Etant donné que les entrées du fichier d'amorce sont encapsulées, les premières entrées de la section du journal d'archivage du fichier d'amorce peuvent être plus récentes que les entrées à la fin. Regardez la combinaison de la date et de l'heure et comparez leurs âges. Ne partez pas du principe que vous pouvez supprimer toutes les entrées avant l'entrée du journal d'archivage contenant le minimum de LOGRBA.

Supprimez les fichiers. Si les archives sont sur bande, effacez les bandes. S'ils se trouvent sur une unité de stockage à accès direct, exécutez un utilitaire z/OS pour supprimer chaque fichier. Ensuite, si vous souhaitez que le fichier d'amorce répertorie uniquement les volumes d'archivage existants, utilisez l'utilitaire d'inventaire du journal des modifications (CSQJU003) pour supprimer les entrées des volumes supprimés. Pour voir un exemple, consultez [«Modifications apportées aux journaux d'archivage»](#), à la page 395.

L'effet de la dérivation des journaux

Les transactions à exécution longue peuvent entraîner des enregistrements de journal d'unité de travail qui couvrent des fichiers journaux. IBM MQ gère ce scénario à l'aide de la dérivation de journal, une technique qui déplace les enregistrements de journal afin d'optimiser la quantité de données de journal conservées et l'heure de redémarrage du gestionnaire de files d'attente.

Lorsqu'une unité d'oeuvre est considérée comme longue, une représentation de chaque enregistrement de journal est consignée plus loin dans le journal. Cette opération est appelée *dérivation de journal*. Il est décrit plus en détail dans [Fichiers journaux](#).

Le gestionnaire de files d'attente utilise ces enregistrements de journal shunted au lieu des enregistrements originaux après un incident, afin de garantir l'intégrité de l'unité de travail. Il y a deux avantages à cela:

- la quantité de données de journal à conserver pour la coordination de l'unité de travail est réduite
- moins de données de journal doivent être parcourues lors du redémarrage du gestionnaire de files d'attente, de sorte que le gestionnaire de files d'attente est redémarré plus rapidement

Les enregistrements de journal shuntés ne contiennent pas suffisamment d'informations pour les opérations de reprise sur incident lié aux supports.

Les données contenues dans le journal sont utilisées à deux fins distinctes: la récupération des supports et la coordination des unités de travail. En cas de défaillance d'un support qui affecte une structure d'unité de couplage ou un ensemble de pages, le gestionnaire de files d'attente peut récupérer le support au point de défaillance en restaurant une copie précédente et en le mettant à jour à l'aide des données contenues dans le journal. L'activité persistante effectuée dans une unité de travail est enregistrée dans le journal de sorte qu'en cas de défaillance, elle peut être annulée ou des verrous peuvent être récupérés sur les ressources modifiées. La quantité de données de journal que vous devez conserver pour activer la reprise du gestionnaire de files d'attente est affectée par ces deux éléments.

Pour la reprise sur incident lié au support, vous devez conserver suffisamment de données de journal pour pouvoir effectuer une reprise sur incident lié au support à partir de la copie de support la plus récente au moins et pour pouvoir effectuer une reprise sur incident. (Votre site peut stipuler la possibilité de récupérer à partir de sauvegardes plus anciennes.) Pour l'intégrité de l'unité de travail, vous devez conserver les données de journal de vos unités de travail en cours ou en attente de validation les plus anciennes.

Pour vous aider à gérer le système, le gestionnaire de files d'attente détecte les anciennes unités de travail à chaque archivage des journaux et les signale dans les messages CSQJ160 et CSQJ161. Une tâche interne lit les informations du journal des unités de travail pour ces anciennes unités de travail et les réécrit sous une forme plus succincte à la position en cours dans le journal. Le message CSQR026 indique quand cela s'est produit. La commande MQSC DISPLAY USAGE TYPE (DATASET) peut également vous aider à gérer la conservation des données de journal. La commande signale 3 éléments d'informations de reprise:

1. la quantité du journal qui doit être conservée pour la récupération de l'unité de travail
2. la quantité du journal à conserver pour la reprise sur support des ensembles de pages
3. pour un gestionnaire de files d'attente dans un groupe de partage de files d'attente, la quantité de journal à conserver pour la reprise sur support des structures d'unité de couplage

Pour chacun de ces éléments, une tentative est effectuée pour mapper les données de journal les plus anciennes requises dans un fichier. À mesure que de nouvelles unités de travail commencent et s'arrêtent, nous nous attendrions à ce que (1) passe à une position plus récente dans le journal. S'il n'est pas déplacé, les messages d'unité de travail à exécution longue vous avertissent qu'il y a un problème. (2) se rapporte à la récupération des supports de l'ensemble de pages si le gestionnaire de files d'attente devait être arrêté maintenant et redémarré. Il ne sait pas quand vous avez sauvegardé vos ensembles de pages pour la dernière fois, ni quelle sauvegarde vous devrez peut-être utiliser en cas d'échec d'un ensemble de pages. Il se déplace normalement vers une position plus récente dans le journal lors du traitement des points de contrôle à mesure que les modifications contenues dans les pools de mémoire tampon sont écrites dans les ensembles de pages. Dans (3), le gestionnaire de files d'attente connaît les sauvegardes de structure d'unité de couplage effectuées sur ce gestionnaire de files d'attente ou sur d'autres gestionnaires de files d'attente du groupe de partage de files d'attente. Toutefois, la récupération de la structure d'unité de couplage nécessite une fusion des données de journal de tous les gestionnaires de files d'attente du groupe de partage de files d'attente qui ont interagi avec la structure d'unité de couplage depuis la dernière sauvegarde. Cela signifie que les données de journal sont identifiées par un numéro de séquence d'enregistrement de journal (ou LRSN), qui est basé sur l'horodatage et donc applicable à l'ensemble du groupe de partage de files d'attente plutôt qu'une adresse relative en octets qui serait différente sur les différents gestionnaires de files d'attente du groupe de partage de files d'attente. Il se déplace normalement vers une position plus récente dans le journal lorsque des commandes BACKUP CFSTRUCT sont exécutées sur ce gestionnaire de files d'attente ou sur d'autres gestionnaires de files d'attente du groupe de partage de files d'attente.

Réinitialisation du journal du gestionnaire de files d'attente

Cette rubrique explique comment réinitialiser le journal du gestionnaire de files d'attente.

Vous ne devez pas autoriser l'adresse relative en octets de journal du gestionnaire de files d'attente à passer de la fin de la plage d'adresses relatives en octets de journal à 0, car cela entraîne une indisponibilité du gestionnaire de files d'attente et toutes les données persistantes deviennent irrécupérables. La fin de l'adresse relative en octets de journal est soit une valeur de FFFFFFFFFFFFFFFF (si des adresses relatives en octets de 6 octets sont utilisées), soit une valeur de FFFFFFFFFFFFFFFF (si des adresses relatives en octets de 8 octets sont utilisées).

Le gestionnaire de files d'attente émet des messages [CSQI045I](#), [CSQI046E](#), [CSQI047E](#), [CSQJ031Det](#) [CSQJ032E](#) pour indiquer que la plage de journaux utilisée est importante et que vous devez planifier une action pour éviter une indisponibilité non planifiée.

Le gestionnaire de files d'attente s'arrête avec le code anomalie [00D10257](#) lorsque la valeur de l'adresse relative en octets atteint FFF800000000 (si des adresses relatives en octets de journal de 6 octets sont utilisées) ou FFFFFFFC00000000 (si des adresses relatives en octets de journal de 8 octets sont utilisées).

Si des adresses relatives en octets de journal de 6 octets sont utilisées, envisagez de convertir le gestionnaire de files d'attente pour qu'il utilise des adresses relatives en octets de journal de 8 octets au lieu de réinitialiser le journal du gestionnaire de files d'attente, en suivant le processus décrit dans [Implémentation de l'adresse relative en octets de journal de plus grande taille](#). La conversion d'un gestionnaire de files d'attente pour utiliser des adresses relatives en octets de journal de 8 octets nécessite une indisponibilité plus courte que la réinitialisation du journal et augmente la durée avant que vous n'ayez à réinitialiser le journal.

Le message [CSQJ034I](#), émis lors de l'initialisation du gestionnaire de files d'attente, indique la fin de la plage d'adresses relatives en octets de journal pour le gestionnaire de files d'attente tel que configuré et peut être utilisé pour déterminer si des adresses relatives en octets de journal de 6 octets ou de 8 octets sont utilisées.

La procédure à suivre pour réinitialiser le journal du gestionnaire de files d'attente est la suivante:

1. Résolvez les unités d'oeuvre non résolues. Le nombre d'unités de travail non résolues est affiché au démarrage du gestionnaire de files d'attente dans le message CSQR005I en tant que nombre d'unités de travail en attente de validation. A chaque point de contrôle et à l'arrêt du gestionnaire de files d'attente, le gestionnaire de files d'attente émet automatiquement la commande

DISPLAY CONN(*) TYPE(CONN) ALL WHERE(UOWSTATE EQ UNRESOLVED) pour fournir des informations sur les unités d'oeuvre non résolues.

Pour plus d'informations sur la résolution des unités de récupération, voir [How in-doubt units of recovery are resolved](#) . Le recours ultime consiste à utiliser la commande **RESOLVE INDOUBT MQSC** pour résoudre manuellement les unités de récupération en attente de validation.

2. Arrêtez le gestionnaire de files d'attente proprement.

Vous pouvez utiliser **STOP QMGR** ou **STOP QMGR MODE(FORCE)** car ces deux commandes permettent de vider les pages modifiées des pools de mémoire tampon vers les ensembles de pages.

3. Si un gestionnaire de files d'attente fait partie d'un groupe de partage de files d'attente, effectuez des sauvegardes CFSTRUCT sur d'autres gestionnaires de files d'attente pour toutes les structures du groupe de partage de files d'attente. Cela garantit que les sauvegardes les plus récentes ne figurent pas dans le journal de ce gestionnaire de files d'attente et que le journal de ce gestionnaire de files d'attente n'est pas requis pour la reprise CFSTRUCT.
4. Définissez les nouveaux journaux et le fichier d'amorce à l'aide de CSQJU003 (voir [L'utilitaire d'inventaire des journaux de modifications](#) pour plus d'informations sur l'utilisation de l'utilitaire d'inventaire des journaux de modifications).
5. Exécutez **CSQUTIL RESETPAGE** sur tous les ensembles de pages de ce gestionnaire de files d'attente (voir [Copie d'une page et réinitialisation du journal](#) pour plus d'informations sur l'utilisation de cette fonction). Notez que les rôles d'ensemble de pages peuvent être réinitialisés indépendamment, de sorte que plusieurs travaux simultanés (par exemple, un par ensemble de pages) peuvent être soumis afin de réduire le temps écoulé pour cette étape.
6. Redémarrez le gestionnaire de files d'attente

Concepts associés

«Implémentation de l'adresse relative en octets de journal de plus grande taille», à la page 389

Avant IBM MQ 8.0, IBM MQ for z/OS utilisait une adresse relative en octets de journal de 6 octets pour identifier l'emplacement des données dans le journal. Depuis la IBM MQ 8.0, la longueur de l'adresse relative en octets du journal peut être de 8 octets, ce qui augmente la période avant que vous n'ayez à réinitialiser le journal.


Implémentation de l'adresse relative en octets de journal de plus grande taille

Avant IBM MQ 8.0, IBM MQ for z/OS utilisait une adresse relative en octets de journal de 6 octets pour identifier l'emplacement des données dans le journal. Depuis la IBM MQ 8.0, la longueur de l'adresse relative en octets du journal peut être de 8 octets, ce qui augmente la période avant que vous n'ayez à réinitialiser le journal.

Cette nouvelle fonction doit être explicitement activée. Voir [Planification de l'augmentation de la plage de journalisation maximale adressable](#) pour plus de détails sur la planification de l'activation de l'adresse relative en octets de journal de 8 octets.

Exécutez les instructions suivantes, dans l'ordre indiqué, pour activer l'adresse relative en octets de journal de 8 octets sur un seul gestionnaire de files d'attente IBM MQ for z/OS :

1. Activez les nouvelles fonctions IBM MQ 8.0 à l'aide de [OPMODE](#).

 Pour les gestionnaires de files d'attente d'un groupe de partage de files d'attente, il n'est pas nécessaire d'effectuer une indisponibilité totale du groupe de partage de files d'attente. Vous pouvez arrêter chaque gestionnaire de files d'attente à tour de rôle, l'activer pour OPMODE = (NEWFUNC,800) ou OPMODE = (NEWFUNC,900) et le redémarrer.

CD

Une fois que tous les gestionnaires de files d'attente du groupe de partage de files d'attente sont en cours d'exécution avec OPMODE = (NEWFUNC,800) ou OPMODE = (NEWFUNC,900), effectuez les étapes suivantes pour chaque gestionnaire de files d'attente du groupe de partage de files d'attente jusqu'à ce que tous les gestionnaires de files d'attente soient en cours d'exécution avec le nouveau fichier d'amorce.

2. Allouez de nouveaux fichiers d'amorce avec des attributs similaires au fichier d'amorce en cours. Vous pouvez personnaliser l'exemple CSQ4BSDS et supprimer toute instruction non pertinente, ou vous pouvez utiliser votre JCL existant, mais remplacez le nom du fichier d'amorce par quelque chose comme ++HLQ++ .NEW .BSDS01.

Remarques :

- a. Vérifiez les attributs de votre nouveau fichier d'amorce. Le seul attribut qui peut changer est la taille du fichier d'amorce.
 - b. Le nouveau fichier d'amorce contient davantage de données que le fichier d'amorce en cours. Par conséquent, vous devez vous assurer que les nouveaux fichiers sont alloués avec un espace disponible suffisant. Pour connaître les valeurs recommandées lors de la définition d'un nouveau fichier d'amorce, voir [Planification de votre environnement de journalisation](#) et les rubriques associées.
3. Arrêtez le gestionnaire de files d'attente proprement.
 4. Exécutez l' [utilitaire de conversion de fichier d'amorce \(CSQJUCNV\)](#) pour convertir le fichier d'amorce existant en nouveaux fichiers d'amorce. Cette opération prend généralement quelques secondes.
Votre fichier d'amorce existant ne sera pas modifié au cours de ce processus, et vous pouvez l'utiliser pour l'initialisation du gestionnaire de files d'attente en cas d'échec de la conversion.
 5. Renommez le fichier d'amorce en cours pour qu'il devienne l'ancien fichier d'amorce et le nouveau fichier d'amorce pour qu'il devienne le fichier d'amorce en cours, de sorte que les nouveaux fichiers soient utilisés lors du prochain redémarrage du gestionnaire de files d'attente. Vous pouvez utiliser la commande DFSMS Access Method Services ALTER, par exemple:

```
ALTER '++HLQ++.BSDS01' NEWNAME('++HLQ++.OLD.BSDS01')  
ALTER '++HLQ++.NEW.BSDS01' NEWNAME('++HLQ++.BSDS01')
```

Veillez également à exécuter des commandes pour renommer les parties données et index du cluster VSAM.

6. Redémarrez le gestionnaire de files d'attente. Il doit démarrer dans le même délai que lors de l'utilisation de l'adresse relative en octets de journal de 6 octets.

Si le gestionnaire de files d'attente ne redémarre pas correctement en raison d'un échec d'accès au fichier d'amorce converti, essayez d'identifier la cause de l'échec, résolvez le problème et relancez l'opération. Si nécessaire, contactez le centre de support IBM pour obtenir de l'aide.

Si nécessaire, la modification peut être annulée à ce stade par:

- a. Renommez le fichier d'amorce en cours pour qu'il devienne le nouveau fichier d'amorce.
- b. Renommez l'ancien fichier d'amorce pour qu'il devienne le fichier d'amorce en cours.
- c. Redémarrage du gestionnaire de files d'attente.

Une fois que le gestionnaire de files d'attente a été redémarré avec succès avec le fichier d'amorce converti, ne tentez pas de démarrer le gestionnaire de files d'attente à l'aide de l'ancien fichier d'amorce.

7. Le message [CSQJ034I](#) est émis lors de l'initialisation du gestionnaire de files d'attente pour indiquer la fin de l'adresse relative en octets du journal pour le gestionnaire de files d'attente tel que configuré. Vérifiez que la fin de la plage d'adresses relative en octets de journal affichée est FFFFFFFFFFFFFFFF. Indiquez que l'adresse relative en octets de journal de 8 octets est en cours d'utilisation.

Remarque : CD Pour activer une adresse relative en octets de journal de 8 octets sur un nouveau gestionnaire de files d'attente IBM MQ 9.0, vous devez d'abord créer un fichier d'amorce de

format de version 1 vide et l'utiliser comme entrée pour l'utilitaire de conversion de fichier d'amorce afin de générer un fichier d'amorce de format de version 2. Pour plus d'informations sur la manière dont vous exécutez ce processus, voir [Création des fichiers d'amorçage et de journalisation](#).

Information associée

[Planification de l'augmentation de la plage de données de journal adressable maximale](#)

[Adresse relative en octets de journal de plus grande taille](#)

[Utilitaire de conversion du fichier d'amorce \(CSQJUCNV\)](#)

Gestion du fichier d'amorçage (BSDS)

Le fichier d'amorçage (BSDS) est utilisé pour référencer des fichiers journaux et des enregistrements de journal. Utilisez cette rubrique pour comprendre comment examiner, modifier et récupérer le fichier d'amorce.

Pour plus d'informations, voir [The bootstrap data set](#).

Cette rubrique décrit les tâches impliquées dans la gestion du fichier d'amorçage. Il contient les sections suivantes:

- [«Découvrir ce que contient le fichier d'amorce»](#), à la page 391
- [«Modification du fichier d'amorce»](#), à la page 392
- [«Récupération du fichier d'amorce»](#), à la page 397

Découvrir ce que contient le fichier d'amorce

Vous pouvez utiliser l'utilitaire d'impression de mappe de journal (CSQJU004) pour examiner le contenu du fichier d'amorce.

L'utilitaire d'impression de mappe de journal (CSQJU004) est un utilitaire de traitement par lots qui répertorie les informations stockées dans le fichier d'amorce. Pour obtenir des instructions sur son exécution, voir [L'utilitaire d'impression de mappe de journaux](#).

Le fichier d'amorce contient:

- [Horodatages](#)
- [Statut du fichier journal actif](#)

Horodatages dans le fichier d'amorce

La sortie de l'utilitaire d'impression de mappe de journal affiche les horodatages, qui sont utilisés pour enregistrer la date et l'heure de divers événements système, qui sont stockés dans le fichier d'amorce.

Les horodatages suivants sont inclus dans la section d'en-tête du rapport:

Horodatage système

Indique la date et l'heure de la dernière mise à jour du fichier d'amorce. L'horodatage du fichier d'amorce peut être mis à jour lorsque:

- Le gestionnaire de files d'attente démarre.
- Le seuil d'écriture est atteint lors des activités d'écriture de journal. Selon le nombre de mémoires tampon de sortie que vous avez spécifié et le taux d'activité du système, le fichier d'amorce peut être mis à jour plusieurs fois par seconde ou ne pas être mis à jour pendant plusieurs secondes, minutes ou même heures. Pour plus de détails sur le seuil d'écriture, voir le paramètre WRTHRS de la macro CSQ6LOGP dans [Utilisation de CSQ6LOGP](#).
- IBM MQ passe en mode de fichier d'amorce unique à partir de son mode de fichier d'amorce double normal en raison d'une erreur. Cela peut se produire lorsqu'une demande d'obtention, d'insertion, de point, de mise à jour ou de suppression d'un enregistrement de fichier d'amorce échoue. Lorsque cette erreur se produit, IBM MQ met à jour l'horodatage dans le fichier

d'amorce restant pour forcer une non-concordance d'horodatage avec le fichier d'amorce désactivé.

HORODATAGE DE L'UTILITAIRE

Date et heure auxquelles le contenu du fichier d'amorce a été modifié par l'utilitaire d'inventaire du journal des modifications (CSQJU003).

Les horodatages suivants sont inclus dans la partie des fichiers journaux actifs et d'archivage du rapport:

Date du journal actif

Date à laquelle l'entrée de journal active a été créée dans le fichier d'amorce, c'est-à-dire lors de l'exécution de la commande CSQJU003 NEWLOG.

Heure du journal actif

Heure à laquelle l'entrée de journal active a été créée dans le fichier d'amorce, c'est-à-dire lors de l'exécution de la commande CSQJU003 NEWLOG.

Date du journal d'archivage

Date à laquelle l'entrée de journal d'archivage a été créée dans le fichier d'amorce, c'est-à-dire lorsque le fichier CSQJU003 NEWLOG a été créé ou que l'archivage lui-même a été effectué.

Heure du journal d'archivage

Heure à laquelle l'entrée de journal d'archivage a été créée dans le fichier d'amorce, c'est-à-dire lorsque le fichier CSQJU003 NEWLOG a été créé ou que l'archivage lui-même a été effectué.

Statut du fichier journal actif

Le fichier d'amorce enregistre le statut d'un fichier journal actif comme suit:

NEW

Le fichier a été défini mais n'a jamais été utilisé par IBM MQ, ou le journal a été tronqué à un point avant la première utilisation du fichier. Dans les deux cas, les valeurs d'adresse relative en octets de début et de fin de fichier sont réinitialisées à zéro.

REUTILISABLE

Soit le fichier a été défini mais n'a jamais été utilisé par IBM MQ, soit le fichier a été déchargé. Dans la sortie de la mappe de journal d'impression, la valeur d'adresse relative en octets de début du dernier fichier RÉUTILISABLE est égale à la valeur d'adresse relative en octets de début du dernier fichier journal d'archivage.

NON REUTILISABLE

Le fichier contient des enregistrements qui n'ont pas été déchargés.

ARRETE

Le processeur de déchargement a rencontré une erreur lors de la lecture d'un enregistrement et cet enregistrement n'a pas pu être obtenu à partir de l'autre copie du journal actif.

Tronqué

L'un ou l'autre :

- Une erreur d'E-S s'est produite et IBM MQ a arrêté d'écrire dans ce fichier. Le fichier journal actif est déchargé, en commençant par l'adresse relative en octets de début et en continuant jusqu'au dernier segment d'enregistrement valide dans le fichier journal actif tronqué. L'adresse relative en octets du dernier segment d'enregistrement valide est inférieure à l'adresse relative en octets de fin du fichier journal actif. La consignation est basculée sur le prochain jeu de données de journal actif disponible et se poursuit sans interruption.

ou

- Une fonction ARCHIVE LOG a été appelée, qui a tronqué le journal actif.

Le statut apparaît dans la sortie de l'utilitaire d'impression de mappe de journaux.

Modification du fichier d'amorce

Vous n'avez pas besoin d'effectuer des étapes spéciales pour conserver le fichier d'amorce à jour avec les enregistrements des événements de consignation car IBM MQ le fait automatiquement.

Toutefois, vous pouvez être amené à modifier le fichier d'amorce si vous effectuez l'une des opérations suivantes:

- Ajoutez d'autres jeux de données de journaux actifs.
- Copiez les fichiers journaux actifs dans les nouveaux fichiers alloués, par exemple, lorsque des allocations de journaux actifs plus importantes sont fournies.
- Déplacez les fichiers journaux vers d'autres périphériques.
- Récupérez un fichier d'amorce endommagé.
- Supprimez les fichiers d'archivage obsolètes.

Vous pouvez modifier le fichier d'amorce en exécutant l'utilitaire de modification de l'inventaire des journaux (CSQJU003). N'exécutez cet utilitaire que lorsque le gestionnaire de files d'attente est inactif ou que vous risquez d'obtenir des résultats incohérents. L'action de l'utilitaire est contrôlée par des instructions du fichier SYSIN. Cette section présente plusieurs exemples. Pour obtenir des instructions complètes, voir [Utilitaire d'inventaire du journal des modifications](#).

Vous pouvez copier un fichier journal actif uniquement lorsque le gestionnaire de files d'attente est inactif car IBM MQ attribue les fichiers journaux actifs comme étant exclusifs (DISP = OLD) au démarrage du gestionnaire de files d'attente.

Modifications apportées aux journaux actifs

Cette rubrique explique comment modifier les journaux actifs à l'aide du fichier d'amorce.

Vous pouvez ajouter, supprimer et enregistrer des entrées dans le fichier d'amorce pour les journaux actifs à l'aide de l'utilitaire de modification du journal. Des exemples uniquement sont affichés ici ; remplacez les noms de fichier affichés par ceux que vous souhaitez utiliser. Pour plus de détails sur l'utilitaire, voir [The change log inventory utility](#).

Pour plus d'informations, voir les sections suivantes:

- [Ajout d'entrées d'enregistrement au fichier d'amorce](#)
- [Suppression des informations relatives au fichier journal actif du fichier d'amorce](#)
- [Enregistrement d'informations sur le fichier journal dans le fichier d'amorce](#)
- [Augmentation de la taille du journal actif](#)
- [Utilisation de CSQJUFMT](#)

Ajout d'entrées d'enregistrement au fichier d'amorce

Si un journal actif a été marqué comme "arrêté", il n'est pas réutilisé pour la consignation ; cependant, il continue d'être utilisé pour la lecture. Utilisez les services de méthode d'accès pour définir de nouveaux fichiers journaux actifs, puis utilisez l'utilitaire d'inventaire du journal des modifications pour enregistrer les nouveaux fichiers dans le fichier d'amorce. Par exemple, utilisez :

```
NEWLOG DSNAMES=MQM111.LOGCOPY1.DS10,COPY1
NEWLOG DSNAMES=MQM111.LOGCOPY2.DS10,COPY2
```

Si vous copiez le contenu d'un ancien fichier journal actif dans le nouveau, vous pouvez également indiquer la plage d'adresses relatives en octets et les horodatages de début et de fin sur la fonction NEWLOG.

Suppression d'informations sur le fichier journal actif du fichier d'amorce

Pour supprimer des informations sur un fichier journal actif du fichier d'amorce, vous pouvez utiliser:

```
DELETE DSNAMES=MQM111.LOGCOPY1.DS99
DELETE DSNAMES=MQM111.LOGCOPY2.DS99
```

Enregistrement d'informations sur le fichier journal dans le fichier d'amorce

Pour enregistrer des informations sur un fichier journal actif existant dans le fichier d'amorce, utilisez:

```
NEWLOG DSNNAME=MQM111.LOGCOPY1.DS10,COPY2,STARTIME=19930212205198,  
ENDTIME=19930412205200,STARTRBA=6400,ENDRBA=94FF
```

Vous devrez peut-être insérer un enregistrement contenant ce type d'informations dans le fichier d'amorce pour les raisons suivantes:

- L'entrée du fichier a été supprimée, mais elle est à nouveau nécessaire.
- Vous copiez le contenu d'un fichier journal actif dans un autre fichier.
- Vous récupérez le fichier d'amorce à partir d'une copie de sauvegarde.

Augmentation de la taille du journal actif

Il existe deux méthodes pour réaliser ce processus.

1. Lorsque le gestionnaire de files d'attente est actif:
 - a. Définissez de nouveaux ensembles de données de journal plus volumineux à l'aide de JCL.
 - b. Ajoutez les nouveaux fichiers journaux au gestionnaire de files d'attente actif à l'aide de la commande MQSC DEFINE LOG.
 - c. Utilisez la commande MQSC ARCHIVE LOG pour déplacer le journal actif en cours afin qu'il soit un nouveau journal plus volumineux.
 - d. Attendez que l'archivage du fichier journal actif plus petit soit terminé.
 - e. Arrêtez le gestionnaire de files d'attente à l'aide de l'utilitaire CSQJU003 pour supprimer les anciens journaux actifs de petite taille.
 - f. Redémarrez le gestionnaire de files d'attente.
2. Lorsque le gestionnaire de files d'attente est inactif:
 - a. Arrêtez le gestionnaire de files d'attente. Cette étape est requise car IBM MQ alloue tous les fichiers journaux actifs pour son utilisation exclusive lorsqu'il est actif.
 - b. Utilisez Access Method Services ALTER avec l'option NEWNAME pour renommer vos fichiers journaux actifs.
 - c. Utilisez les services de méthode d'accès DEFINE pour définir des fichiers journaux actifs plus volumineux.

En réutilisant les anciens noms de fichier, vous n'avez pas besoin d'exécuter l'utilitaire d'inventaire du journal des modifications pour établir de nouveaux noms dans les fichiers d'amorce. Les anciens noms de fichier et les plages RBA correctes se trouvent déjà dans les fichiers d'amorce.
 - d. Utilisez Access Method Services REPRO pour copier les anciens ensembles de données (renommés) dans leurs nouveaux ensembles de données appropriés.

Remarque : Cette étape peut prendre beaucoup de temps, de sorte que votre entreprise peut être hors service pendant cette période.

- e. Démarrez le gestionnaire de files d'attente.

Si tous vos fichiers journaux ont la même taille, votre système sera plus cohérent et plus efficace sur le plan opérationnel. Si les fichiers journaux ne sont pas de la même taille, il est plus difficile de suivre les journaux de votre système et l'espace peut donc être gaspillé.

Utilisation de CSQJUFMT

N'exécutez pas le format CSQJUFMT lorsque vous augmentez la taille d'un journal actif.

Si vous exécutez CSQJUFMT (afin d'améliorer les performances lors de la première écriture du gestionnaire de files d'attente dans le nouveau journal actif), vous recevez les messages suivants:

```
IEC070I 203-204,XS95GTLX,REPRO02,OUTPUT,B857,SPMG02, 358
IEC070I MG.W.MG4E.LOGCOPY1.DS02,MG.W.MG4E.LOGCOPY1.DS02.DATA,
IDC3302I ACTION ERROR ON MG.W.MG4E.LOGCOPY1.DS02
IDC3351I ** VSAM I/O RETURN CODE IS 28 - RPLFDBWD = X'2908001C'
IDC31467I MAXIMUM ERROR LIMIT REACHED.

IDC0005I NUMBER OF RECORDS PROCESSED WAS 0
```

En outre, si vous utilisez Access Method Services REPRO, veillez à définir un nouveau journal vide.

Si vous utilisez REPRO pour copier l'ancien fichier (renommé) dans son nouveau fichier respectif, la valeur par défaut est NOREPLACE.

Cela signifie que REPRO ne remplace pas un enregistrement qui se trouve déjà dans le fichier désigné. Lorsque le formatage est effectué sur le fichier, la valeur RBA est réinitialisée. Le résultat net est un fichier qui n'est pas vide après le formatage.



Modifications apportées aux journaux d'archivage

Cette rubrique explique comment modifier les journaux d'archivage.

Vous pouvez ajouter, supprimer et modifier le mot de passe des entrées du fichier d'amorce pour les journaux d'archivage. Des exemples uniquement sont affichés ici ; remplacez les noms de fichier affichés par ceux que vous souhaitez utiliser. Pour plus de détails sur l'utilitaire, voir [The change log inventory utility](#).

- [Ajout d'un journal d'archivage](#)
- [Suppression d'un journal d'archivage](#)
- [Modification du mot de passe d'un journal d'archivage](#)

Ajout d'un journal d'archivage

Lorsque la récupération d'un objet dépend de la lecture d'un fichier d'archivage existant, le fichier d'amorce doit contenir des informations sur ce fichier afin que IBM MQ puisse le trouver. Pour enregistrer des informations sur un fichier d'archivage existant dans le fichier d'amorce, utilisez:

```
NEWLOG DSNAME=CSQARC1.ARCHLOG1.E00021.T2205197.A0000015,COPY1VOL=CSQV04,
UNIT=TAPE,STARTRBA=3A190000,ENDRBA=3A1F0FFF,CATALOG=NO
```

Suppression d'un journal d'archivage

Pour supprimer la totalité d'un fichier d'archivage sur un ou plusieurs volumes, utilisez:

```
DELETE DSNAME=CSQARC1.ARCHLOG1.E00021.T2205197.A0000015,COPY1VOL=CSQV04
```

Modification du mot de passe d'un journal d'archivage

Si vous modifiez le mot de passe d'un fichier d'archivage existant, vous devez également modifier les informations du fichier d'amorce.

1. Répertoriez le fichier d'amorce à l'aide de l'utilitaire d'impression de mappe de journal.
2. Supprimez l'entrée du fichier journal d'archivage avec le mot de passe modifié, à l'aide de la fonction DELETE de l'utilitaire CSQJU003 (voir la rubrique [Utilitaire de modification de l'inventaire du journal](#)).

3. Nommez le fichier comme pour un nouveau fichier d'archivage. Utilisez la fonction NEWLOG de l'utilitaire CSQJU003 (voir la rubrique [The change log inventory utility](#)) et indiquez le nouveau mot de passe, les adresses relatives en octets de début et de fin et les numéros de série de volume (qui se trouvent dans la sortie de l'utilitaire d'impression de mappe de journal, voir [The print log map utility](#)).

Pour modifier le mot de passe des nouveaux fichiers journaux d'archivage, utilisez:

```
ARCHIVE PASSWORD= password
```

Pour arrêter de placer des mots de passe dans les nouveaux fichiers journaux d'archivage, utilisez:

```
ARCHIVE NOPASSWD
```

Remarque : N'utilisez l'utilitaire ARCHIVE que si vous ne disposez pas d'un gestionnaire de sécurité externe.



Modification du qualificatif de haut niveau (HLQ) pour les journaux et le fichier d'amorce
Utilisez cette rubrique pour comprendre la procédure à suivre pour modifier le qualificatif de haut niveau (HLQ).

Avant de commencer

Vous devez arrêter le gestionnaire de files d'attente normalement avant de copier les journaux ou les fichiers dans les nouveaux fichiers. Cela permet de s'assurer que les données sont cohérentes et qu'aucune reprise n'est nécessaire lors du redémarrage.

Pourquoi et quand exécuter cette tâche

Cette tâche fournit des informations sur la modification du qualificatif de haut niveau pour les journaux et le fichier d'amorce. Pour cela, procédez comme suit :

Procédure

1. Exécutez l'utilitaire d'impression de journal CSQJU004 pour enregistrer les informations du fichier journal. Ces informations sont nécessaires ultérieurement.
2. Vous pouvez :
 - a) exécution de la sauvegarde et de la restauration DSS avec changement de nom dans le journal et les fichiers d'amorce à renommer, ou
 - b) utiliser AMS DEFINE et REPRO pour créer les fichiers HLQ et copier les données des anciens fichiers.
3. Modifiez les procédures MSTR et CHIN pour qu'elles pointent vers les nouveaux fichiers.
4. Supprimez les anciennes informations de journal dans la nouvelle copie du fichier d'amorce à l'aide de CSQJU003.
5. Définissez les nouveaux fichiers journaux dans le nouveau fichier d'amorce à l'aide de la fonction NEWLOG de CSQJU003.
Conservez toutes les informations sur chaque journal de la même manière, à l'exception du qualificatif de haut niveau.
6. Le nouveau fichier d'amorce doit refléter les mêmes informations que celles qui ont été enregistrées pour les anciens journaux dans l'ancien fichier d'amorce.
Le HLQ devrait être la seule chose qui a changé.

Que faire ensuite

Comparez la sortie CSQJU004 de l'ancien et du nouveau fichier d'amorce pour vous assurer qu'ils sont EXACTEMENT identiques (à l'exception des fichiers HLQ) avant de démarrer le gestionnaire de files d'attente.

Remarque : Il faut être prudent lors de l'exécution de ces opérations. Des actions incorrectes peuvent entraîner des situations irrémédiables. Vérifiez la sortie de l'utilitaire PRINT LOG MAP UTILITY et assurez-vous que toutes les informations nécessaires à la reprise ou au redémarrage ont été incluses.

Récupération du fichier d'amorce

Si IBM MQ fonctionne en mode de fichier d'amorce double et qu'un fichier d'amorce est endommagé, ce qui force IBM MQ à passer en mode de fichier d'amorce unique, IBM MQ continue à fonctionner sans problème (jusqu'au prochain redémarrage).

Pour rétablir l'environnement en mode de fichier d'amorce double:

1. Utilisez les services de méthode d'accès pour renommer ou supprimer le fichier d'amorce endommagé et pour définir un nouveau fichier d'amorce portant le même nom que le fichier d'amorce endommagé. Des exemples d'instructions de contrôle sont disponibles dans le travail CSQ4BREC dans thlqual.SCSQPROC.
2. Exécutez la commande IBM MQ RECOVER BSDS pour effectuer une copie du fichier d'amorce valide dans le fichier nouvellement alloué et pour rétablir le mode de fichier d'amorce double.

Si IBM MQ fonctionne en mode de fichier d'amorce unique et que le fichier d'amorce est endommagé, ou si IBM MQ fonctionne en mode de fichier d'amorce double et que les deux fichiers d'amorce sont endommagés, le gestionnaire de files d'attente s'arrête et ne redémarre pas tant que les fichiers d'amorce ne sont pas réparés. Dans ce cas :

1. Localisez le fichier d'amorce associé au fichier d'archivage le plus récent. Le nom de fichier du journal d'archivage le plus récent apparaît dans le journal des travaux lors de la dernière occurrence du message CSQJ003I, qui indique que le traitement du déchargement a abouti. En vue de la suite de cette procédure, il est recommandé de conserver un journal de toutes les archives ayant abouti et notées par ce message:
 - Si les journaux d'archivage se trouvent sur une unité de stockage à accès direct, le fichier d'amorce est alloué sur n'importe quelle unité de stockage à accès direct disponible. Le nom du fichier d'amorce est similaire au nom de fichier d'archivage correspondant ; modifiez uniquement la première lettre du dernier qualificatif, de A à B, comme dans l'exemple suivant:
Nom du journal d'archivage
CSQ.ARCHLOG1. **A** 0000001
Nom de la copie du fichier d'amorce
CSQ.ARCHLOG1. **B** 0000001
 - Si les journaux d'archivage sont sur bande, le fichier d'amorce est le premier fichier du premier volume de journal d'archivage. Le fichier d'amorce n'est pas répété sur les volumes ultérieurs.
2. Si le fichier d'archivage le plus récent ne possède pas de copie du fichier d'amorce (par exemple, en raison d'une erreur lors de son déchargement), recherchez une copie antérieure du fichier d'amorce à partir du traitement de déchargement précédent.
3. Renommez les fichiers d'amorce *endommagés* à l'aide de la commande Access Method Services ALTER avec l'option NEWNAME. Si vous souhaitez supprimer un fichier d'amorce endommagé, utilisez la commande Access Method Services DELETE. Pour chaque fichier d'amorce endommagé, utilisez les services de méthode d'accès pour définir un nouveau fichier d'amorce en tant que fichier de remplacement. Le travail CSQ4BREC dans thlqual.SCSQPROC contient des instructions de contrôle Access Method Services pour définir un nouveau fichier d'amorce.
4. Utilisez la commande Access Method Services REPRO pour copier le fichier d'amorce du journal d'archivage vers l'un des fichiers d'amorce de remplacement que vous avez définis à l'étape «3», à la page 397. Ne copiez pas de données dans le deuxième fichier d'amorce de remplacement, comme indiqué à l'étape «5», à la page 399.

a. Imprimez le contenu du fichier d'amorce de remplacement.

Utilisez l'utilitaire d'impression de mappe de journal (CSQJU004) pour imprimer le contenu du fichier d'amorce de remplacement. Cela vous permet de passer en revue le contenu du fichier d'amorce de remplacement avant de poursuivre votre travail de reprise.

b. Mettez à jour l'inventaire des fichiers d'archivage dans le fichier d'amorce de remplacement.

Examinez la sortie de l'utilitaire d'impression de mappe de journal et vérifiez que le fichier d'amorce de remplacement ne contient pas d'enregistrement du journal d'archivage à partir duquel le fichier d'amorce a été copié. Si le fichier d'amorce de remplacement est une ancienne copie, son inventaire risque de ne pas contenir tous les fichiers d'archivage créés plus récemment. L'inventaire du fichier d'amorce des fichiers d'archivage doit être mis à jour pour refléter l'inventaire du sous-système en cours.

Utilisez l'utilitaire de modification de l'inventaire des journaux (CSQJU003) NEWLOG pour mettre à jour le fichier d'amorce de remplacement, en ajoutant un enregistrement du journal d'archivage à partir duquel le fichier d'amorce a été copié. Si le fichier journal d'archivage est protégé par mot de passe, utilisez l'option PASSWORD de la fonction NEWLOG. De plus, si le fichier d'archivage est catalogué, vérifiez que l'option CATALOG de la fonction NEWLOG est correctement définie sur CATALOG=YES. Utilisez l'instruction NEWLOG pour ajouter des fichiers d'archivage supplémentaires créés après la copie du fichier d'amorce.

c. Mettez à jour les mots de passe dans le fichier d'amorce de remplacement.

Le fichier d'amorce contient des mots de passe pour les fichiers journaux d'archivage et pour les fichiers journaux actifs. Pour vous assurer que les mots de passe du fichier d'amorce de remplacement reflètent les mots de passe en cours utilisés par votre installation, utilisez la fonction d'utilitaire ARCHIVE de l'inventaire du journal des modifications avec l'option PASSWORD.

d. Mettez à jour l'inventaire des fichiers journaux actifs dans le fichier d'amorce de remplacement.

Dans des circonstances inhabituelles, votre installation peut avoir ajouté, supprimé ou renommé des fichiers journaux actifs depuis que le fichier d'amorce a été copié. Dans ce cas, le fichier d'amorce de remplacement ne reflète pas le nombre ou les noms réels des fichiers journaux actifs actuellement utilisés par votre installation.

Si vous devez supprimer un fichier journal actif de l'inventaire de journal BSDS de remplacement, utilisez la fonction DELETE de l'utilitaire de modification d'inventaire de journal.

Si vous devez ajouter un fichier journal actif à l'inventaire du journal du fichier d'amorce de remplacement, utilisez la fonction NEWLOG de l'utilitaire de modification de l'inventaire du journal. Vérifiez que la plage d'adresses relatives en octets est correctement indiquée dans la fonction NEWLOG. Si le fichier journal actif est protégé par mot de passe, utilisez l'option PASSWORD.

Si vous devez renommer un fichier journal actif dans l'inventaire du journal du fichier d'amorce de remplacement, utilisez la fonction DELETE de l'utilitaire de modification de l'inventaire du journal, suivie de la fonction NEWLOG. Vérifiez que la plage d'adresses relatives en octets est correctement indiquée dans la fonction NEWLOG. Si le fichier journal actif est protégé par mot de passe, utilisez l'option PASSWORD.

e. Mettez à jour les plages d'adresses de journal actives dans le fichier d'amorce de remplacement.

Par la suite, lorsque le gestionnaire de files d'attente redémarre, il compare les adresses relatives en octets des fichiers journaux actifs répertoriés dans le fichier d'amorce avec les adresses relatives en octets trouvées dans les fichiers journaux actifs réels. Si les rôles ne sont pas d'accord, le gestionnaire de files d'attente ne redémarre pas. Le problème est amplifié lorsqu'une ancienne copie du fichier d'amorce est utilisée. Pour résoudre ce problème, utilisez l'utilitaire d'inventaire du journal des modifications (CSQJU003) pour ajuster les rôles trouvés dans le fichier d'amorce à l'aide des rôles dans les fichiers journaux actifs réels. Pour ce faire, procédez comme suit:

- Utilisation de l'utilitaire d'impression des enregistrements de journal (CSQ1LOGP) pour imprimer un rapport récapitulatif du fichier journal actif. Affiche les adresses relatives en octets de début et de fin.

- Comparaison des plages d'adresses relatives en octets réelles avec les plages d'adresses relatives en octets que vous venez d'imprimer, lorsque les adresses relatives en octets de tous les fichiers journaux actifs sont connues.

Si les plages d'adresses relatives en octets sont égales pour tous les fichiers journaux actifs, vous pouvez passer à l'étape de reprise suivante sans aucun travail supplémentaire.

Si les plages de l'adresse relative en octets ne sont pas égales, ajustez les valeurs du fichier d'amorce pour refléter les valeurs réelles. Pour chaque fichier journal actif dont la plage d'adresses relatives en octets doit être ajustée, utilisez la fonction DELETE de l'utilitaire de modification de l'inventaire du journal pour supprimer le fichier journal actif de l'inventaire dans le fichier d'amorce de remplacement. Utilisez ensuite la fonction NEWLOG pour redéfinir le fichier journal actif dans le fichier d'amorce. Si les fichiers journaux actifs sont protégés par mot de passe, utilisez l'option PASSWORD de la fonction NEWLOG.

- f. Si seuls deux fichiers journaux actifs sont spécifiés pour chaque copie du journal actif, IBM MQ peut rencontrer des difficultés lors du redémarrage du gestionnaire de files d'attente. Le problème peut se produire lorsque l'un des fichiers journaux actifs est saturé et n'a pas été déchargé, alors que le second fichier journal actif est sur le point de se remplir. Dans ce cas, ajoutez un nouveau fichier journal actif pour chaque copie du journal actif et définissez chaque nouveau fichier journal actif dans l'inventaire du journal BSDS de remplacement.

Utilisez la commande Access Method Services DEFINE pour définir un nouveau fichier journal actif pour chaque copie du journal actif et utilisez la fonction NEWLOG de l'utilitaire de modification de l'inventaire du journal pour définir les nouveaux fichiers journaux actifs dans le fichier d'amorce de remplacement. Il n'est pas nécessaire de spécifier les plages d'adresses relatives en octets dans l'instruction NEWLOG. Toutefois, si les fichiers journaux actifs sont protégés par mot de passe, utilisez l'option PASSWORD de la fonction NEWLOG. Des exemples d'instructions de contrôle permettant d'accomplir cette tâche se trouvent dans le travail CSQ4LREC dans thlqual.SCSQPROC.

5. Copiez le fichier d'amorce mis à jour dans le deuxième nouveau fichier d'amorce. Les fichiers d'amorce sont maintenant identiques.

Utilisez l'utilitaire d'impression de mappe de journal (CSQJU004) pour imprimer le contenu du deuxième fichier d'amorce de remplacement à ce stade.

6. Pour plus d'informations sur les actions à effectuer si vous avez perdu votre fichier journal actif en cours, voir [Problèmes liés aux journaux actifs](#).
7. Redémarrez le gestionnaire de files d'attente à l'aide du nouveau fichier d'amorce. IBM MQ détermine l'adresse relative en octets en cours et les journaux actifs à archiver.

Gestion des ensembles de pages

Cette rubrique explique comment gérer les ensembles de pages associés à un gestionnaire de files d'attente.

Cette rubrique décrit comment ajouter, copier et généralement gérer les ensembles de pages associés à un gestionnaire de files d'attente. Il contient les sections suivantes:

- [«Comment modifier le qualificatif de haut niveau \(HLQ\) pour les ensembles de pages»](#), à la page 400
- [«Comment ajouter un ensemble de pages à un gestionnaire de files d'attente»](#), à la page 400
- [«Que faire lorsque l'un de vos ensembles de pages est saturé»](#), à la page 401
- [«Comment équilibrer les charges sur les ensembles de pages»](#), à la page 401
- [Comment augmenter la taille d'un ensemble de pages](#)
- [«Comment réduire un ensemble de pages»](#), à la page 405
- [«Comment réintroduire un ensemble de pages»](#), à la page 405
- [«Comment sauvegarder et récupérer des ensembles de pages»](#), à la page 406

- [«Comment supprimer des ensembles de pages», à la page 410](#)
- [«Comment sauvegarder et restaurer des files d'attente à l'aide de CSQUTIL», à la page 411](#)

Voir [Ensembles de pages](#) pour obtenir une description des ensembles de pages, des classes de stockage, des mémoires tampon et des pools de mémoire tampon, ainsi que certaines considérations relatives aux performances qui s'appliquent.

Comment modifier le qualificatif de haut niveau (HLQ) pour les ensembles de pages

Cette tâche fournit des informations sur la façon de modifier le qualificatif de haut niveau pour les ensembles de pages. Pour effectuer cette tâche, procédez comme suit:

1. Définissez les nouveaux ensembles de pages HLQ.
2. Si l'allocation de taille est identique à celle des anciens ensembles de pages, copiez l'ensemble de pages existant à l'aide de REPRO dans les nouveaux ensembles de pages HLQ vides. Si vous augmentez la taille des ensembles de pages, utilisez la fonction FORMAT de CSQUTIL pour formater l'ensemble de pages de destination. Pour plus d'informations, voir [Formatage des ensembles de pages \(FORMAT\)](#).
3. Utilisez la fonction COPYPAGE de CSQUTIL pour copier tous les messages de l'ensemble de pages source vers l'ensemble de pages de destination. Pour plus d'informations, voir [Développement d'un ensemble de pages \(COPYPAGE\)](#).
4. Modifiez l'instruction de définition de données CSQP00xx dans la procédure du gestionnaire de files d'attente pour qu'elle pointe vers les nouveaux ensembles de pages HLQ.

Redémarrez le gestionnaire de files d'attente et vérifiez les modifications apportées aux ensembles de pages.

Comment ajouter un ensemble de pages à un gestionnaire de files d'attente

Cette description suppose que vous disposez d'un gestionnaire de files d'attente déjà en cours d'exécution. Vous devrez peut-être ajouter un ensemble de pages si, par exemple, votre gestionnaire de files d'attente doit faire face à de nouvelles applications utilisant de nouvelles files d'attente.

Pour ajouter un nouvel ensemble de pages, procédez comme suit:

1. Définissez et formatez le nouvel ensemble de pages. Vous pouvez utiliser l'exemple de JCL dans `thlqlal.SCSQPROC(CSQ4PAGE)` comme base. Pour plus d'informations, voir [Formatage des ensembles de pages \(FORMAT\)](#).
Prenez soin de ne pas formater les ensembles de pages en cours d'utilisation, sauf si c'est ce que vous souhaitez. Si tel est le cas, utilisez l'option FORCE de la fonction d'utilitaire FORMAT.
2. Utilisez la commande DEFINE PSID avec l'option DSN pour associer l'ensemble de pages à un pool de mémoire tampon.
3. Ajoutez les définitions de classe de stockage appropriées pour votre ensemble de pages en exécutant les commandes DEFINE STGCLASS.
4. Si vous le souhaitez, pour documenter la configuration de votre gestionnaire de files d'attente:
 - a. Ajoutez le nouvel ensemble de pages à la procédure de tâche démarrée pour votre gestionnaire de files d'attente.
 - b. Ajoutez une définition pour le nouvel ensemble de pages à votre fichier d'initialisation CSQINP1 .
 - c. Ajoutez une définition pour la nouvelle classe de stockage à votre membre de fichier d'initialisation CSQ4INYP .

Pour plus de détails sur les commandes DEFINE PSID et DEFINE STGCLASS, voir [DEFINE PSID](#) et [DEFINE STGCLASS](#).

Que faire lorsque l'un de vos ensembles de pages est saturé

Pour plus d'informations sur l'utilisation des ensembles de pages, utilisez la commande IBM MQ DISPLAY USAGE. Par exemple, la commande

```
DISPLAY USAGE PSID(03)
```

affiche l'état en cours de l'ensemble de pages 03. Cela vous indique le nombre de pages libres de cet ensemble de pages.

Si vous avez défini des domaines secondaires pour vos ensembles de pages, ils sont développés dynamiquement chaque fois qu'ils sont remplis. Finalement, toutes les extensions secondaires sont utilisées ou aucun espace disque supplémentaire n'est disponible. Dans ce cas, une application reçoit le code retour MQRC_STORAGE_MEDIUM_FULL.

Si une application reçoit le code retour MQRC_STORAGE_MEDIUM_FULL d'un appel MQI, cela indique clairement qu'il ne reste pas assez d'espace sur l'ensemble de pages. Si le problème persiste ou est susceptible de se reproduire, vous devez faire quelque chose pour le résoudre.

Vous pouvez aborder ce problème de plusieurs manières:

- Equilibre la charge entre les ensembles de pages en déplaçant des files d'attente d'un ensemble de pages à un autre.
- Développez l'ensemble de pages. Voir [«Comment augmenter la taille d'un ensemble de pages»](#), à la page 403 pour des instructions.
- Redéfinissez l'ensemble de pages de sorte qu'il puisse s'étendre au-delà de 4 Go jusqu'à une taille maximale de 64 Go. Pour obtenir des instructions, voir [Définition d'un ensemble de pages supérieur à 4 Go](#).

Comment équilibrer les charges sur les ensembles de pages

L'équilibrage de charge sur les ensembles de pages signifie le déplacement des messages associés à une ou plusieurs files d'attente d'un ensemble de pages vers un autre ensemble de pages moins utilisé. Utilisez cette technique s'il n'est pas pratique de développer l'ensemble de pages.

Pour identifier les files d'attente qui utilisent un ensemble de pages, utilisez les commandes IBM MQ appropriées. Par exemple, pour identifier les files d'attente mappées à l'ensemble de pages 02, commencez par identifier les classes d'archivage mappées à l'ensemble de pages 02, à l'aide de la commande suivante:

```
DISPLAY STGCLASS(*) PSID(02)
```

Utilisez ensuite la commande suivante pour déterminer quelles files d'attente utilisent quelle classe d'archivage:

```
DISPLAY QUEUE(*) TYPE(QLOCAL) STGCLASS
```

Déplacement d'une file d'attente non partagée

Pour déplacer des files d'attente et leurs messages d'un ensemble de pages à un autre, utilisez la commande MQSC MOVE QLOCAL (décrite dans [MOVE QLOCAL](#)). Une fois que vous avez identifié la ou les files d'attente que vous souhaitez déplacer vers un nouvel ensemble de pages, procédez comme suit pour chacune de ces files d'attente:

1. Vérifiez que la file d'attente que vous souhaitez déplacer n'est utilisée par aucune application (c'est-à-dire que les valeurs IPPROCS et OPPROCS de la commande DISPLAY QSTATUS sont nulles) et qu'elle ne comporte aucun message non validé (la valeur UNCOM de la commande DISPLAY QSTATUS est NO).

Remarque : Le seul moyen de s'assurer que cet état se poursuit est de modifier temporairement l'autorisation de sécurité de la file d'attente. Pour plus d'informations, voir [Profils de sécurité de file d'attente](#).

Si vous ne pouvez pas effectuer cette opération, les étapes ultérieures de cette procédure risquent d'échouer si les applications commencent à utiliser la file d'attente en dépit des mesures de précaution telles que la définition de PUT (DISABLED). Toutefois, les messages ne peuvent jamais être perdus par cette procédure.

2. Empêchez les applications d'insérer des messages dans la file d'attente en cours de déplacement en modifiant la définition de la file d'attente pour désactiver MQPUT s. Remplacez la définition de file d'attente par PUT (DISABLED).
3. Définissez une file d'attente temporaire avec les mêmes attributs que la file d'attente en cours de déplacement, à l'aide de la commande suivante:

```
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED)
```

Remarque : Si cette file d'attente temporaire existe déjà à partir d'une exécution précédente, supprimez-la avant de procéder à la définition.

4. Déplacez les messages dans la file d'attente temporaire à l'aide de la commande suivante:

```
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
```

5. Supprimez la file d'attente que vous déplacez à l'aide de la commande suivante:

```
DELETE QLOCAL(Queue_To_Move)
```

6. Définissez une nouvelle classe de stockage qui est mappée à l'ensemble de pages requis, par exemple:

```
DEFINE STGCLASS(NEW) PSID(nn)
```

Ajoutez la nouvelle définition de classe de stockage aux fichiers CSQINP2 prêts pour le prochain redémarrage du gestionnaire de files d'attente.

7. Redéfinissez la file d'attente que vous déplacez en modifiant l'attribut de classe de stockage:

```
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) STGCLASS(NEW)
```

Lorsque la file d'attente est redéfinie, elle est basée sur la file d'attente temporaire créée à l'étape «3», à la page 402.

8. Remplacez les messages dans la nouvelle file d'attente à l'aide de la commande suivante:

```
MOVE QLOCAL(TEMP) TOQLOCAL(Queue_To_Move)
```

9. La file d'attente créée à l'étape «3», à la page 402 n'est plus requise. Utilisez la commande suivante pour la supprimer:

```
DELETE QL(TEMP_QUEUE)
```

10. Si la file d'attente en cours de déplacement a été définie dans les fichiers CSQINP2, modifiez l'attribut STGCLASS de la commande DEFINE QLOCAL appropriée dans les fichiers CSQINP2. Ajoutez le mot clé REPLACE pour remplacer la définition de file d'attente existante.

La Figure 46, à la page 403 présente une extraction d'un travail d'équilibrage de charge.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=th1qua1.SCSQANLE,DISP=SHR
// DD DSN=th1qua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_To_Move) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_To_Move) PUT(ENABLED) GET(ENABLED)
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_To_Move)
DEFINE STGCLASS(NEW) PSID(2)
DEFINE QL(Queue_To_Move) LIKE(TEMP_QUEUE) STGCLASS(NEW)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_To_Move)
DELETE QL(TEMP_QUEUE)
/*
```

Figure 46. Extraction à partir d'un travail d'équilibrage de charge pour un ensemble de pages

Comment augmenter la taille d'un ensemble de pages

Vous pouvez allouer initialement un ensemble de pages supérieur à 4 Go. Voir [Définition d'un ensemble de pages supérieur à 4 Go](#)

Un ensemble de pages peut être défini pour être développé automatiquement lorsqu'il est plein en spécifiant EXPAND (SYSTEM) ou EXPAND (USER). Si votre ensemble de pages a été défini avec EXPAND (NONE), vous pouvez le développer de deux manières:

- Modifiez sa définition pour autoriser l'extension automatique. Voir [Modification d'un ensemble de pages pour autoriser l'extension automatique](#)
- Créez un ensemble de pages plus grand et copiez les messages de l'ancien ensemble de pages vers le nouvel ensemble. Voir [Déplacement de messages vers un nouvel ensemble de pages plus grand](#)

Définition d'un ensemble de pages d'une taille supérieure à 4 Go

IBM MQ peut utiliser une page d'une taille maximale de 64 Go, à condition que le fichier soit défini avec une capacité d'adressage étendue pour VSAM. L'adressabilité étendue est un attribut conféré par une classe de données SMS. Dans l'exemple de JCL suivant, la classe de gestion EXTENDED est définie sur SMS avec 'Extended addressability'. Si votre ensemble de pages existant n'est pas actuellement défini comme ayant une capacité d'adressage étendue, utilisez la méthode suivante pour migrer vers un fichier de format de capacité d'adressage étendue.

1. Arrêtez le gestionnaire de files d'attente.

- Utilisez Access Method Services pour renommer l'ensemble de pages existant.
- Définissez un ensemble de pages de destination, de la même taille que l'ensemble de pages existant, mais avec DATACLAS (EXTENDED).

Remarque : Les fichiers au format étendu doivent être gérés par SMS. Les mécanismes de demande de format étendu pour les fichiers VSAM sont les suivants:

- Utilisation d'une classe de données dont la valeur DSNTYPE est EXT et du sous-paramètre R ou P pour indiquer une valeur requise ou préférée.
- Codage DSNTYPE=EXTREQ (le format étendu est requis) ou DSNTYPE=EXTPREF (le format étendu est préféré) dans l'instruction DD.
- Codage du paramètre LIKE= dans l'instruction de définition de données pour faire référence à un fichier de format étendu existant.

Pour plus d'informations, voir [Restrictions relatives à la définition de fichiers au format étendu](#).

- Utilisez la fonction COPYPAGE de CSQUTIL pour copier tous les messages de l'ensemble de pages source vers l'ensemble de pages de destination. Pour plus d'informations, voir [Développement d'un ensemble de pages \(COPYPAGE\)](#).
- Redémarrez le gestionnaire de files d'attente.
- Modifiez l'ensemble de pages pour utiliser l'extension du système, afin de lui permettre de continuer à croître au-delà de son allocation actuelle.

Le JCL suivant montre des exemples de commandes Access Method Services:

```
//S1 EXEC PGM=IDCAMS
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
ALTER 'VICY.CSQ1.PAGE01' -
NEWNAME('VICY.CSQ1.PAGE01.OLD')
ALTER 'VICY.CSQ1.PAGE01.DATA' -
NEWNAME('VICY.CSQ1.PAGE01.DATA.OLD')
DEFINE CLUSTER (NAME('VICY.CSQ1.PAGE01') -
MODEL('VICY.CSQ1.PAGE01.OLD') -
DATACLAS(EXTENDED))
/*
```

Modification d'un ensemble de pages pour autoriser l'extension automatique

Utilisez la commande ALTER PSID avec les options EXPAND (USER) ou EXPAND (SYSTEM). Voir [ALTER PSID](#) et [Extension d'un ensemble de pages \(COPYPAGE\)](#) pour obtenir des informations générales sur l'extension des ensembles de pages.

Déplacement de messages vers un nouvel ensemble de pages plus grand

Cette technique implique l'arrêt et le redémarrage du gestionnaire de files d'attente. Cette opération supprime tous les messages non persistants qui ne se trouvent pas dans les files d'attente partagées au moment du redémarrage. Si vous avez des messages non persistants que vous ne souhaitez pas supprimer, utilisez l'équilibrage de charge à la place. Pour plus de détails, voir [«Comment équilibrer les charges sur les ensembles de pages»](#), à la page 401. Dans cette description, l'ensemble de pages que vous souhaitez développer est appelé ensemble de pages *source* ; le nouvel ensemble de pages plus grand est appelé ensemble de pages *destination*.

Procédez comme suit :

- Arrêtez le gestionnaire de files d'attente.
- Définissez l'ensemble de pages de destination en vous assurant qu'il est plus grand que l'ensemble de pages source, avec une valeur d'extension secondaire plus élevée.
- Utilisez la fonction FORMAT de CSQUTIL pour formater l'ensemble de pages de destination. Pour plus d'informations, voir [Formatage des ensembles de pages \(FORMAT\)](#).

4. Utilisez la fonction COPYPAGE de CSQUTIL pour copier tous les messages de l'ensemble de pages source vers l'ensemble de pages de destination. Pour plus d'informations, voir [Développement d'un ensemble de pages \(COPYPAGE\)](#).
5. Redémarrez le gestionnaire de files d'attente à l'aide de l'ensemble de pages de destination en effectuant l'une des opérations suivantes:
 - Modifiez la procédure de tâche démarrée du gestionnaire de files d'attente pour référencer l'ensemble de pages de destination.
 - Utilisez Access Method Services pour supprimer l'ensemble de pages source, puis renommez l'ensemble de pages de destination en lui attribuant le même nom que celui de l'ensemble de pages source.

Avertissement :

Avant de supprimer un ensemble de pages IBM MQ, assurez-vous d'avoir effectué les copies de sauvegarde requises.

Comment réduire un ensemble de pages

Empêchez tous les utilisateurs, à l'exception de l'administrateur IBM MQ, d'utiliser le gestionnaire de files d'attente. Par exemple, en modifiant les paramètres de sécurité d'accès.

Si vous disposez d'un ensemble de pages de grande taille qui est principalement vide (comme indiqué par la commande DISPLAY USAGE), vous souhaitez peut-être réduire sa taille. La procédure à suivre consiste à utiliser les fonctions COPY, FORMAT et LOAD de CSQUTIL (voir [Programme utilitaire IBM MQ](#)). Cette procédure ne fonctionne pas pour l'ensemble de pages zéro (0), car il n'est pas pratique de réduire la taille de cet ensemble de pages ; la seule façon de le faire est de réinitialiser votre gestionnaire de files d'attente (voir «[Réinitialisation d'un gestionnaire de files d'attente](#)», à la page 432). La condition préalable à cette procédure est de tenter de supprimer tous les utilisateurs du système afin que toutes les unités de travail soient complètes et que les ensembles de pages soient cohérents.

1. Utilisez la commande STOP QMGR avec l'attribut QUIESCE ou FORCE pour arrêter le gestionnaire de files d'attente.
2. Exécutez la fonction SCOPY de CSQUTIL avec l'option PSID pour copier toutes les données de message du grand ensemble de pages et les sauvegarder dans un fichier séquentiel.
3. Définissez un nouvel ensemble de données d'ensemble de pages plus petit pour remplacer l'ensemble de pages de grande taille.
4. Exécutez la fonction FORMAT TYPE (NEW) de CSQUTIL sur l'ensemble de pages que vous avez créé à l'étape «3», à la page 405.
5. Redémarrez le gestionnaire de files d'attente à l'aide de l'ensemble de pages créé à l'étape «3», à la page 405.
6. Exécutez la fonction LOAD de CSQUTIL pour recharger tous les messages sauvegardés à l'étape «2», à la page 405.
7. Autorisez tous les utilisateurs à accéder au gestionnaire de files d'attente.
8. Supprimez l'ancien ensemble de grandes pages.

Comment réintroduire un ensemble de pages

Dans certains scénarios, il est utile de pouvoir remettre en ligne un ancien ensemble de pages dans le gestionnaire de files d'attente. A moins qu'une action spécifique ne soit effectuée, lorsque l'ancien ensemble de pages est mis en ligne, le gestionnaire de files d'attente reconnaît que l'adresse relative en octets de récupération de l'ensemble de pages stocké dans l'ensemble de pages lui-même et dans les enregistrements de point de contrôle est ancienne et démarre automatiquement la récupération de support de l'ensemble de pages pour le mettre à jour.

Cette récupération de support ne peut être effectuée qu'au redémarrage du gestionnaire de files d'attente et prend probablement beaucoup de temps, en particulier si les journaux d'archivage conservés sur bande doivent être lus. Toutefois, normalement, dans ce cas, l'ensemble de pages a été mis hors ligne

pendant la période intermédiaire et, par conséquent, le journal ne contient aucune information relative à la récupération de l'ensemble de pages.

Les trois options suivantes sont disponibles:

Permet d'effectuer une reprise complète des supports.

1. Arrêtez le gestionnaire de files d'attente.
2. Vérifiez que des définitions sont disponibles pour l'ensemble de pages dans la procédure de tâche démarrée pour le gestionnaire de files d'attente et dans le fichier d'initialisation CSQINP1 .
3. Redémarrez le gestionnaire de files d'attente.

Autoriser la destruction de tous les messages de l'ensemble de pages.

Ce choix est utile lorsqu'un ensemble de pages est hors ligne depuis longtemps (quelques mois, par exemple) et qu'il a été décidé de le réutiliser à des fins différentes.

1. Formatez l'ensemble de pages à l'aide de la fonction FORMAT de CSQUTIL avec l'option TYPE (NEW).
2. Ajoutez des définitions pour l'ensemble de pages à la fois à la procédure de tâche démarrée pour le gestionnaire de files d'attente et au fichier d'initialisation CSQINP1 .
3. Redémarrez le gestionnaire de files d'attente.

L'utilisation de l'option TYPE (NEW) pour le formatage efface le contenu en cours de l'ensemble de pages et indique au gestionnaire de files d'attente d'ignorer les informations d'historique dans le point de contrôle sur l'ensemble de pages.

Mettez l'ensemble de pages en ligne en évitant le processus de reprise sur incident lié au support.

Utilisez cette technique uniquement si vous êtes certain que l'ensemble de pages a été déconnecté depuis un arrêt correct du gestionnaire de files d'attente. Cette option est la plus appropriée lorsque l'ensemble de pages a été déconnecté pendant une courte période, généralement en raison de problèmes opérationnels tels qu'une sauvegarde en cours d'exécution lors du démarrage du gestionnaire de files d'attente.

1. Formatez l'ensemble de pages à l'aide de la fonction FORMAT de CSQUTIL avec l'option TYPE (REPLACE).
2. Ajoutez à nouveau l'ensemble de pages dans le gestionnaire de files d'attente de manière dynamique à l'aide de la commande DEFINE PSID avec l'option DSN ou autorisez son ajout lors du redémarrage du gestionnaire de files d'attente.

L'utilisation de l'option TYPE (REPLACE) permet de vérifier que l'ensemble de pages a été correctement fermé par le gestionnaire de files d'attente et de le marquer de sorte que la reprise sur incident lié au support ne soit pas effectuée. Aucune autre modification n'est apportée au contenu de l'ensemble de pages.

Comment sauvegarder et récupérer des ensembles de pages

Différents mécanismes sont disponibles pour la sauvegarde et la reprise. Utilisez cette rubrique pour comprendre ces mécanismes.

Cette section décrit les rubriques suivantes:

- [«Création d'un point de récupération pour les ressources non partagées», à la page 407](#)
- [«Sauvegarde des ensembles de pages», à la page 408](#)
- [«Récupération des ensembles de pages», à la page 409](#)
- [Comment supprimer des ensembles de pages](#)

Pour plus d'informations sur la création d'un point de reprise pour les ressources partagées, voir [«Récupération des files d'attente partagées», à la page 416.](#)

Création d'un point de récupération pour les ressources non partagées

IBM MQ peut récupérer des objets et des messages persistants non partagés dans leur état actuel si les deux:

1. Il existe des copies des ensembles de pages d'un point antérieur.
2. Tous les journaux IBM MQ sont disponibles pour effectuer une reprise à partir de ce point.

Il s'agit d'un point de reprise pour les ressources non partagées.

Les objets et les messages sont conservés dans des ensembles de pages. Plusieurs objets et messages provenant de différentes files d'attente peuvent exister dans le même ensemble de pages. A des fins de récupération, les objets et les messages ne peuvent pas être sauvegardés de manière isolée. Par conséquent, un ensemble de pages doit être sauvegardé dans son ensemble pour garantir la récupération correcte des données.

Le journal de reprise IBM MQ contient un enregistrement de tous les messages persistants et des modifications apportées aux objets. Si IBM MQ échoue (par exemple, en raison d'une erreur d'E-S sur un ensemble de pages), vous pouvez récupérer l'ensemble de pages en restaurant la copie de sauvegarde et en redémarrant le gestionnaire de files d'attente. IBM MQ applique les modifications de journal à l'ensemble de pages à partir du point de la copie de sauvegarde.

Il existe deux façons de créer un point de reprise:

Sauvegarde intégrale

Arrêtez le gestionnaire de files d'attente, ce qui force toutes les mises à jour des ensembles de pages.

Cela vous permet de redémarrer à partir du point de récupération, en utilisant uniquement les fichiers d'ensemble de pages sauvegardés et les journaux à partir de ce point.

Sauvegarde partielle

Effectuez des copies de sauvegarde *floue* des ensembles de pages sans arrêter le gestionnaire de files d'attente.

Si vous utilisez cette méthode et que vos journaux associés sont ultérieurement endommagés ou perdus, vous ne pouvez pas utiliser les copies de sauvegarde d'ensemble de pages floues pour effectuer une récupération. En effet, les copies de sauvegarde d'ensemble de pages partielles contiennent une vue incohérente de l'état du gestionnaire de files d'attente et dépendent des journaux disponibles. Si les journaux ne sont pas disponibles, vous devez revenir au dernier groupe de copies de l'ensemble de pages de sauvegarde effectué alors que le sous-système était inactif ([Méthode 1](#)) et accepter la perte de données à partir de ce moment.

Méthode 1: Sauvegarde intégrale

Cette méthode implique l'arrêt du gestionnaire de files d'attente. Cela force toutes les mises à jour sur les ensembles de pages afin que les ensembles de pages soient dans un état cohérent.

1. Arrêtez toutes les applications IBM MQ qui utilisent le gestionnaire de files d'attente (ce qui leur permet de se terminer en premier). Pour ce faire, vous pouvez modifier les paramètres de sécurité d'accès ou de file d'attente, par exemple.
2. Une fois toutes les activités terminées, affichez et résolvez les unités de récupération en attente de validation. (Utilisez les commandes `DISPLAY CONN` et `RESOLVE INDOUBT`, comme décrit dans [DISPLAY CONN](#) et [RESOLVE INDOUBT](#).)

Cela amène les ensembles de pages à un état cohérent ; si vous ne le faites pas, vos ensembles de pages risquent de ne pas être cohérents et vous effectuez une sauvegarde partielle.

3. Exécutez la commande `ARCHIVE LOG` pour vous assurer que les données de journal les plus récentes sont écrites dans les fichiers journaux.
4. Exécutez la commande `STOP QMGR MODE (QUIESCE)`. Enregistrez la valeur d'adresse relative en octets la plus basse dans les messages `CSQI024I` ou `CSQI025I` (voir [CSQI024I](#) et [CSQI025I](#) pour plus d'informations). Vous devez conserver les fichiers journaux à partir de celui indiqué par la valeur de l'adresse relative en octets jusqu'au fichier journal en cours.

5. Effectuez des copies de sauvegarde de tous les ensembles de pages du gestionnaire de files d'attente (voir «Sauvegarde des ensembles de pages», à la page 408).

Méthode 2: Sauvegarde partielle

Cette méthode n'implique pas l'arrêt du gestionnaire de files d'attente. Par conséquent, les mises à jour peuvent se trouver dans des mémoires tampon virtuelles lors du processus de sauvegarde. Cela signifie que les ensembles de pages ne sont pas dans un état cohérent et ne peuvent être utilisés que pour la récupération avec les journaux.

1. Exécutez la commande DISPLAY USAGE TYPE (ALL) et enregistrez la valeur RBA dans les messages CSQI024I ou CSQI025I (voir [CSQI024I](#) et [CSQI025I](#) pour plus d'informations).
2. Effectuez des copies de sauvegarde des ensembles de pages (voir «Sauvegarde des ensembles de pages», à la page 408).
3. Exécutez la commande ARCHIVE LOG pour vous assurer que les données de journal les plus récentes sont écrites dans les fichiers journaux. Pour redémarrer à partir du point de reprise, vous devez conserver les fichiers journaux à partir du fichier journal indiqué par la valeur de l'adresse relative en octets jusqu'au fichier journal en cours.

Sauvegarde des ensembles de pages

Pour récupérer un ensemble de pages, IBM MQ doit savoir jusqu'où il se trouve dans le journal. IBM MQ gère un numéro d'adresse relative en octets de journal dans la page zéro de chaque ensemble de pages, appelé *numéro de séquence du journal de reprise* (LSN). Ce nombre correspond à l'adresse relative en octets de départ dans le journal à partir de laquelle IBM MQ peut récupérer l'ensemble de pages. Lorsque vous sauvegardez un ensemble de pages, ce nombre est également copié.

Si la copie est utilisée ultérieurement pour récupérer l'ensemble de pages, IBM MQ doit avoir accès à tous les enregistrements de journal de cette valeur d'adresse relative en octets à l'adresse relative en octets en cours. Cela signifie que vous devez conserver suffisamment d'enregistrements de journal pour permettre à IBM MQ de récupérer à partir de la copie de sauvegarde la plus ancienne d'un ensemble de pages que vous avez l'intention de conserver.

Utilisez la fonction ADRDSSU COPY pour copier les ensembles de pages.

Pour plus d'informations, voir la documentation [COPY DATASET Command Syntax for Logical Data Set](#).

Exemple :

```
//STEP2 EXEC PGM=ADRDSSU,REGION=6M
//SYSPRINT DD SYSOUT=H
//SYSIN DD *
COPY -
DATASET(INCLUDE(SCENDATA.MQPA.PAGESET.*)) -
RENAMEU(SCENDATA.MQPA.PAGESET.** ,SCENDATA.MQPA.BACKUP1.** ) -
SPHERE -
REPUNC -
FASTREPLICATION(PREF ) -
CANCELERROR -
TOL(ENQF)
/*
//
```

Si vous copiez l'ensemble de pages alors que le gestionnaire de files d'attente est en cours d'exécution, vous devez utiliser un utilitaire de copie qui copie d'abord la page zéro de l'ensemble de pages. Si vous ne le faites pas, vous risquez d'endommager les données de votre ensemble de pages.

Si le processus de développement dynamique d'un ensemble de pages est interrompu, par exemple en cas de perte d'alimentation du système, vous pouvez toujours utiliser ADRDSSU pour effectuer une sauvegarde d'un ensemble de pages.

Si vous effectuez un LISTCAT ENT('page set data set name') ALLOCAccess Method Services IDCAMS, vous verrez que HI-ALLOC-RBA est supérieur à HI-USED-RBA.

La prochaine fois que cet ensemble de pages est rempli, il est étendu à nouveau, si possible, et les pages entre l'adresse relative en octets maximale utilisée et l'adresse relative en octets maximale allouée sont utilisées, avec une autre nouvelle extension.

Sauvegarde des définitions d'objet

Vous devez également sauvegarder des copies de vos définitions d'objet. Pour ce faire, utilisez la fonction MAKEDEF de la fonction CSQUTIL COMMAND (décrite dans [Emission de commandes pour IBM MQ \(COMMAND\)](#)).

Sauvegardez vos définitions d'objet chaque fois que vous effectuez une copie de sauvegarde de votre gestionnaire de files d'attente et conservez la version la plus récente.

Récupération des ensembles de pages

Si le gestionnaire de files d'attente s'est arrêté en raison d'une défaillance, il peut normalement être redémarré avec toutes les opérations de reprise effectuées pendant le redémarrage. Toutefois, cette récupération n'est pas possible si l'un de vos ensembles de pages ou de fichiers journaux n'est pas disponible. L'étendue de la récupération dépend de la disponibilité des copies de sauvegarde des ensembles de pages et des fichiers journaux.

Pour redémarrer à partir d'un point de reprise, vous devez disposer des éléments suivants:

- Copie de sauvegarde de l'ensemble de pages à récupérer.
- Si vous avez utilisé le processus de sauvegarde "floue" décrit dans [«Méthode 2: Sauvegarde partielle»](#), à la page 408, le fichier journal incluant la valeur d'adresse relative en octets enregistrée, le fichier journal créé par la commande ARCHIVE LOG et tous les fichiers journaux entre ces deux éléments.
- Si vous avez utilisé la sauvegarde intégrale, mais que vous ne disposez pas des fichiers journaux qui suivent ceux de la commande ARCHIVE LOG, vous n'avez **pas** besoin d'exécuter la fonction FORMAT TYPE (REPLACE) de l'utilitaire CSQUTIL sur tous vos ensembles de pages.

Pour récupérer un ensemble de pages dans son état actuel, vous devez également disposer de tous les fichiers journaux et de tous les enregistrements depuis la commande ARCHIVE LOG.

Il existe deux méthodes pour récupérer un ensemble de pages. Pour utiliser l'une ou l'autre de ces méthodes, le gestionnaire de files d'attente doit être arrêté.

Récupération simple

Il s'agit de la méthode la plus simple, qui convient à la plupart des situations de reprise.

1. Supprimez l'ensemble de pages à restaurer à partir de la sauvegarde.
2. Utilisez la fonction ADRDSSU COPY pour récupérer votre ensemble de pages à partir de la copie de sauvegarde.

Vous pouvez également renommer votre copie de sauvegarde avec le nom d'origine ou modifier l'instruction de définition de données CSQP00xx dans votre procédure de gestionnaire de files d'attente pour qu'elle pointe vers votre groupe de pages de sauvegarde. Toutefois, si vous perdez ou endommagez l'ensemble de pages, vous n'aurez plus de copie de sauvegarde à partir de laquelle effectuer la restauration.

3. Redémarrez le gestionnaire de files d'attente.
4. Une fois le gestionnaire de files d'attente redémarré, vous pouvez redémarrer vos applications
5. Rétablissez vos procédures de sauvegarde normales pour la page restaurée.

Reprise avancée

Cette méthode offre des avantages en termes de performances si vous disposez d'un grand ensemble de pages à récupérer ou si l'ensemble de pages a fait l'objet d'une activité importante depuis la dernière copie de sauvegarde. Cependant, elle nécessite plus d'intervention manuelle que la

méthode simple, ce qui peut augmenter le risque d'erreur et le temps nécessaire pour effectuer la récupération.

1. Supprimez et redéfinissez l'ensemble de pages à restaurer à partir de la sauvegarde.
2. Utilisez ADRDSSU pour copier la copie de sauvegarde du jeu de pages dans le nouveau jeu de pages. Définissez votre nouvel ensemble de pages avec une valeur d'extension secondaire afin qu'il puisse être développé de manière dynamique.

Vous pouvez également renommer votre copie de sauvegarde avec le nom d'origine ou modifier l'instruction de définition de données CSQP00xx dans votre procédure de gestionnaire de files d'attente pour qu'elle pointe vers votre groupe de pages de sauvegarde. Toutefois, si vous perdez ou endommagez l'ensemble de pages, vous n'aurez plus de copie de sauvegarde à partir de laquelle effectuer la restauration.

3. Modifiez les définitions CSQINP1 de votre gestionnaire de files d'attente pour que le pool de mémoire tampon associé à l'ensemble de pages récupéré soit le plus grand possible. En rendant le pool de mémoire tampon volumineux, vous pouvez conserver toutes les pages modifiées dans le pool de mémoire tampon et réduire la quantité d'E-S dans l'ensemble de pages.
4. Redémarrez le gestionnaire de files d'attente.
5. Une fois le gestionnaire de files d'attente redémarré, arrêtez-le (à l'aide de la mise au repos), puis redémarrez-le à l'aide de la définition de pool de mémoire tampon normale pour cet ensemble de pages. Une fois ce second redémarrage terminé, vous pouvez redémarrer vos applications
6. Rétablissez vos procédures de sauvegarde normales pour la page restaurée.

Que se passe-t-il lorsque le gestionnaire de files d'attente est redémarré

Lorsque le gestionnaire de files d'attente est redémarré, il applique toutes les modifications apportées à l'ensemble de pages qui sont enregistrées dans le journal, à partir du point de redémarrage de l'ensemble de pages. IBM MQ peut récupérer plusieurs ensembles de pages de cette manière. L'ensemble de pages est développé dynamiquement, si nécessaire, lors de la reprise sur incident lié au support.

Lors du redémarrage, IBM MQ détermine l'adresse relative en octets de journal à partir de laquelle démarrer en prenant la valeur la plus faible parmi les suivantes:

- Numéro de séquence de journal de reprise à partir de l'enregistrement de journal de point de contrôle pour chaque ensemble de pages.
- LSN de récupération à partir de la page zéro dans chaque ensemble de pages.
- Adresse relative en octets de l'unité de récupération incomplète la plus ancienne du système au moment où la sauvegarde a été effectuée.

Toutes les définitions d'objet sont stockées sur l'ensemble de pages zéro. Les messages peuvent être stockés dans n'importe quel ensemble de pages disponible.

Remarque : Le gestionnaire de files d'attente ne peut pas redémarrer si l'ensemble de pages zéro n'est pas disponible.

Comment supprimer des ensembles de pages

Vous supprimez un ensemble de pages à l'aide de la commande DELETE PSID ; voir [DELETE PSID](#) pour plus de détails sur cette commande.

Vous ne pouvez pas supprimer un ensemble de pages qui est toujours référencé par une classe de stockage. Utilisez DISPLAY STGCLASS pour identifier les classes de stockage qui font référence à un ensemble de pages.

Le fichier est libéré de IBM MQ mais n'est pas supprimé. Il reste disponible pour une utilisation ultérieure ou peut être supprimé à l'aide des fonctions z/OS .

Supprimez l'ensemble de pages de la procédure de tâche démarrée pour votre gestionnaire de files d'attente.

Supprimez la définition de l'ensemble de pages de votre fichier d'initialisation CSQINP1 .

Comment sauvegarder et restaurer des files d'attente à l'aide de CSQUTIL

Utilisez cette rubrique comme référence pour plus d'informations sur la sauvegarde et la restauration à l'aide de CSQUTIL.

Vous pouvez utiliser les fonctions de l'utilitaire CSQUTIL pour sauvegarder et restaurer des files d'attente. Pour sauvegarder une file d'attente, utilisez la fonction COPY ou SCOPY pour copier les messages d'une file d'attente dans un fichier. Pour restaurer la file d'attente, utilisez la fonction complémentaire LOAD ou SLOAD. Pour plus d'informations, voir [Programme utilitaire IBM MQ](#).

Gestion des pools de mémoire tampon

Utilisez cette rubrique si vous souhaitez modifier ou supprimer vos pools de mémoire tampon.

Cette rubrique explique comment modifier et supprimer des pools de mémoire tampon. Il contient les sections suivantes:

- [«Comment modifier le nombre de mémoires tampon dans un pool de mémoire tampon»](#), à la page 411
- [«Comment supprimer un pool de mémoire tampon»](#), à la page 412

Les pools de mémoire tampon sont définis lors de l'initialisation du gestionnaire de files d'attente, à l'aide des commandes [DEFINE BUFFPOOL](#) émises à partir du fichier d'entrée d'initialisation CSQINP1. Leurs attributs peuvent être modifiés en réponse à des besoins métier lorsque le gestionnaire de files d'attente est en cours d'exécution, à l'aide des processus détaillés dans cette rubrique. Le gestionnaire de files d'attente enregistre les attributs de pool de mémoire tampon en cours dans les enregistrements de journal de point de contrôle. Ils sont automatiquement restaurés lors du redémarrage suivant du gestionnaire de files d'attente, sauf si la définition du pool de mémoire tampon dans CSQINP1 inclut l'attribut REPLACE.

La commande [DISPLAY USAGE](#) permet d'afficher les attributs de mémoire tampon en cours.

Vous pouvez également définir les pools de mémoire tampon de manière dynamique à l'aide de la commande [DEFINE PSID](#) avec l'option DSN.

Si vous modifiez les pools de mémoire tampon de manière dynamique, vous devez également mettre à jour leurs définitions dans le fichier d'initialisation CSQINP1.

Voir [Planification sur z/OS](#) pour obtenir une description des ensembles de pages, des classes de stockage, des mémoires tampon et des pools de mémoire tampon, ainsi que certaines considérations relatives aux performances qui s'appliquent.

Remarque : Les pools de mémoire tampon utilisent une mémoire importante. Lorsque vous augmentez la taille d'un pool de mémoire tampon ou que vous définissez un nouveau pool de mémoire tampon, assurez-vous que la mémoire disponible est suffisante. Pour plus d'informations, voir [Stockage de l'espace adresse](#).

Comment modifier le nombre de mémoires tampon dans un pool de mémoire tampon

Si un pool de mémoire tampon est trop petit, la condition peut générer le message [CSQP020E](#) sur la console, vous pouvez lui allouer davantage de mémoires tampon à l'aide de la commande ALTER BUFFPOOL, comme suit:

1. Déterminez la quantité d'espace disponible pour les nouvelles mémoires tampon en consultant les messages [CSQY220I](#) dans le journal. L'espace disponible est indiqué en Mo. Comme une mémoire tampon a une taille de 4 Ko, chaque Mo d'espace disponible vous permet d'allouer 256 mémoires

tampon. N'allouez pas tout l'espace disponible aux mémoires tampon, car certaines sont requises pour d'autres tâches.

Si le pool de mémoire tampon utilise des pages fixes de 4 ko, c'est-à-dire que son attribut PAGECLAS est FIXED4KB, vérifiez que la mémoire réelle disponible sur la partition logique est suffisante.

2. Si l'espace disponible signalé est insuffisant, libérez des mémoires tampon d'un autre pool de mémoire tampon à l'aide de la commande

```
ALTER BUFFPOOL(buf-pool-id) BUFFERS(integer)
```

où *buf-pool-id* est le pool de mémoire tampon à partir duquel vous souhaitez récupérer de l'espace et *integer* est le nouveau nombre de mémoires tampon à allouer à ce pool de mémoire tampon, qui doit être inférieur au nombre initial de mémoires tampon qui lui ont été allouées.

3. Ajoutez des mémoires tampon au pool de mémoire tampon que vous souhaitez développer à l'aide de la commande

```
ALTER BUFFPOOL(buf-pool-id) BUFFERS(integer)
```

où *buf-pool-id* est le pool de mémoire tampon à étendre et *integer* est le nouveau nombre de mémoires tampon à allouer à ce pool de mémoire tampon, qui doit être supérieur au nombre initial de mémoires tampon qui lui ont été allouées.

Comment supprimer un pool de mémoire tampon

Lorsqu'un pool de mémoire tampon n'est plus utilisé par des ensembles de pages, supprimez-le pour libérer la mémoire virtuelle qui lui est allouée.

Vous pouvez supprimer un pool de mémoire tampon à l'aide de la commande `DELETE BUFFPOOL`. La commande échoue si des ensembles de pages utilisent ce pool de mémoire tampon.

Pour plus d'informations sur la suppression d'ensembles de pages, voir [«Comment supprimer des ensembles de pages»](#), à la page 410.

Gestion des groupes de partage de files d'attente et des files d'attente partagées sous z/OS

IBM MQ peut utiliser différents types de ressources partagées, par exemple des groupes de partage de files d'attente, des files d'attente partagées et l'unité de couplage. Utilisez cette rubrique pour passer en revue les procédures nécessaires à la gestion de ces ressources partagées.

Cette section contient des informations sur les rubriques suivantes:

- [«Gestion des groupes de partage de files d'attente»](#), à la page 412
- [«Gestion des files d'attente partagées»](#), à la page 416
- [«Gestion des objets de groupe»](#), à la page 422
- [«Gestion de l'unité de couplage»](#), à la page 422

Gestion des groupes de partage de files d'attente

Vous pouvez ajouter ou supprimer un gestionnaire de files d'attente dans un groupe de partage de files d'attente (QSG) et gérer les tables Db2 associées.

Cette rubrique contient des sections sur les tâches suivantes:

- [«Configuration d'un groupe de partage de files d'attente»](#), à la page 413
- [«Ajout d'un gestionnaire de files d'attente à un groupe de partage de files d'attente»](#), à la page 413

- [«Suppression d'un gestionnaire de files d'attente d'un groupe de partage de files d'attente»](#), à la page 415
- [«Suppression d'un groupe de partage de files d'attente des tables Db2»](#), à la page 416
- [«Validation de la cohérence des définitions Db2»](#), à la page 416

Configuration d'un groupe de partage de files d'attente

Chaque groupe de partage de files d'attente possède un nom pouvant comporter jusqu'à quatre caractères. Il doit être unique sur votre réseau et être différent de celui d'un gestionnaire de files d'attente.

Pour configurer un groupe de partage de files d'attente, procédez comme suit:

1. S'il s'agit du premier groupe de partage de files d'attente à utiliser le groupe de partage de données Db2 , [configurez l'environnement Db2](#).
 2. [Configurez l'unité de couplage](#).
 3. Ajoutez le groupe de partage de files d'attente aux tables Db2 . Utilisez la fonction ADD QSG de l'utilitaire de groupe de partage de files d'attente (CSQ5PQSG). Ce programme est décrit dans [Utilitaire de groupe de partage de files d'attente](#). Un exemple est fourni dans `thlqual.SCSQPROC(CSQ45AQS)`.
 4. Ajoutez un gestionnaire de files d'attente au groupe de partage de files d'attente en suivant les étapes de la rubrique [«Ajout d'un gestionnaire de files d'attente à un groupe de partage de files d'attente»](#), à la page 413
 5. Définissez des structures d'application dans IBM MQ en suivant les étapes de la rubrique [«Ajout d'une structure d'unité de couplage»](#), à la page 422.
 6. Si nécessaire, [migrez les files d'attente non partagées vers les files d'attente partagées](#).
 7. Pour la disponibilité, créez des canaux partagés dans et hors du groupe de partage de files d'attente.
 - Pour les connexions au groupe de partage de files d'attente:
 - Configurez un socket VIPA ou un routeur matériel pour répartir la charge de travail entre les gestionnaires de files d'attente disponibles dans le groupe de partage de files d'attente.
 - Définissez un canal récepteur avec QSGDISP (GROUP) pour vous assurer que la définition de canal est disponible sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente.
 - Démarrez un programme d'écoute avec INDISP (GROUP), sur chaque gestionnaire de files d'attente, pour les connexions de canal MCA dans le groupe de partage de files d'attente. Les connexions client au groupe de partage de files d'attente doivent toujours se connecter à un programme d'écoute démarré avec INDISP (QMGR).
 - Modifiez les applications pour qu'elles se connectent à l'aide du nom QSG plutôt qu'à l'aide d'un nom de gestionnaire de files d'attente spécifique.
 - Assurez-vous que les règles d'authentification de canal de tous les gestionnaires de files d'attente du groupe de partage de files d'attente sont identiques pour permettre aux applications de se connecter à n'importe quel gestionnaire de files d'attente du groupe de partage de files d'attente.
 - Pour les connexions en dehors du groupe de partage de files d'attente:
 - Définissez une file d'attente de transmission partagée.
 - Définissez le canal sortant avec QSGDISP (GROUP) et DEFCDISP (SHARED).
- Si vous convertissez un canal existant en canal partagé, vous devrez peut-être exécuter la commande `RESET CHANNEL` avant de démarrer le canal car la file d'attente de synchronisation utilisée par le canal aura changé.

Ajout d'un gestionnaire de files d'attente à un groupe de partage de files d'attente

Un gestionnaire de files d'attente peut être ajouté à un groupe de partage de files d'attente existant.

Notez que :

- Le groupe de partage de files d'attente doit exister pour que vous puissiez y ajouter des gestionnaires de files d'attente.
- Un gestionnaire de files d'attente ne peut être membre que d'un seul groupe de partage de files d'attente.

Pour ajouter un gestionnaire de files d'attente à un groupe de partage de files d'attente, procédez comme suit:

1. Exécutez les tâches de la rubrique Implémentation des contrôles de sécurité ESM pour le groupe de partage de files d'attente afin d'accorder l'accès approprié aux ID utilisateur du gestionnaire de files d'attente et de l'initiateur de canal.
2. Si le groupe de partage de files d'attente possède des structures d'unité de couplage configurées pour décharger des données dans le fichier SMDS, effectuez les tâches de la rubrique Configuration de l'environnement SMDS.
3. Arrêtez le gestionnaire de files d'attente.
4. Utilisez la fonction ADD QMGR de l'utilitaire de groupe de partage de files d'attente (CSQ5PQSG). Ce programme est décrit dans l'utilitaire de groupe de partage de files d'attente. Un exemple est fourni dans thlqual.SCSQPROC(CSQ45AQM).
5. Modifiez votre module de paramètres système pour ajouter des données de groupe de partage de files d'attente:
 - a. Modifiez CSQ6SYSP pour spécifier le paramètre QSGDATA. Pour plus d'informations, voir Utilisation de CSQ6SYSP .
 - b. Assemblez et liez le module de paramètres système. Vous pouvez utiliser un autre nom pour le module de chargement.
 - c. Modifiez votre processus de démarrage pour utiliser le nouveau module.
6. Copiez et personnalisez l'exemple de membre thlqual.SCSQPROC(CSQ4INSS), qui définit les structures d'unité de couplage requises et les files d'attente SYSTEM. Ajoutez le membre personnalisé à la définition de données CSQINP2 dans le JCL de démarrage du gestionnaire de files d'attente.
7. Redémarrez votre gestionnaire de files d'attente à l'aide du module de paramètres système du groupe de partage de files d'attente.
8. Vous pouvez éventuellement effectuer une migration vers des profils de sécurité préfixés par le nom du groupe de partage de files d'attente, à la place du nom du gestionnaire de files d'attente.
9. Si des canaux partagés sont utilisés pour les connexions au groupe de partage de files d'attente, créez des règles d'authentification de canal qui reflètent celles des autres gestionnaires de files d'attente du groupe de partage de files d'attente, afin de permettre aux applications de se connecter à n'importe quel gestionnaire de files d'attente du groupe de partage de files d'attente.
10. Si vous le souhaitez, effectuez l'une des opérations suivantes pour permettre aux applications connectées au gestionnaire de files d'attente dans le groupe de partage de files d'attente d'insérer des messages dans des files d'attente hébergées par d'autres gestionnaires de files d'attente dans le groupe de partage de files d'attente:
 - Activez la mise en file d'attente intra-groupe en exécutant la commande ALTER QMGR IGQ (ENABLED).
 - Définissez les files d'attente de transmission et les canaux vers les autres gestionnaires de files d'attente du groupe de partage de files d'attente. La définition de files d'attente de transmission portant le même nom que les gestionnaires de files d'attente cible évite d'avoir à définir des files d'attente distantes et des alias de gestionnaire de files d'attente.

Remarque : Pour ajouter un gestionnaire de files d'attente à un groupe de partage de files d'attente existant contenant des gestionnaires de files d'attente exécutant des versions antérieures d' IBM MQ, vous devez d'abord appliquer la PTF de coexistence pour la version la plus élevée de IBM MQ dans le groupe à chaque gestionnaire de files d'attente de version antérieure dans le groupe.

Suppression d'un gestionnaire de files d'attente d'un groupe de partage de files d'attente

Vous ne pouvez supprimer un gestionnaire de files d'attente d'un groupe de partage de files d'attente que si les journaux du gestionnaire de files d'attente ne sont pas requis par un autre processus et que tous les fichiers SMDS appartenant au gestionnaire de files d'attente sont vides.

Les journaux sont nécessaires s'ils contiennent:

- Dernière sauvegarde de l'une des structures d'application d'unité de couplage (CF) utilisées par le groupe de partage de files d'attente
- Données requises par un processus de restauration ultérieur, c'est-à-dire que le gestionnaire de files d'attente a utilisé une structure récupérable depuis l'heure décrite par la valeur du dernier intervalle d'exclusion de sauvegarde.

Si l'un de ces points ou les deux s'appliquent ou si un fichier SMDS appartenant au gestionnaire de files d'attente contient des messages, le gestionnaire de files d'attente ne peut pas être supprimé. Pour déterminer les journaux des gestionnaires de files d'attente nécessaires à un processus de restauration ultérieur, utilisez la commande MQSC DISPLAY CFSTATUS avec l'option TYPE (BACKUP) (pour plus de détails sur cette commande, voir [DISPLAY CFSTATUS](#)).

Pour supprimer un gestionnaire de files d'attente d'un groupe de partage de files d'attente, procédez comme suit:

1. Arrêtez toutes les applications connectées au gestionnaire de files d'attente qui ont inséré des messages dans des files d'attente partagées.
2. Résolvez les unités d'oeuvre en attente de validation impliquant ce gestionnaire de files d'attente.
3. Déterminez s'il existe des messages dans un fichier SMDS appartenant au gestionnaire de files d'attente en exécutant la commande DISPLAY USAGE TYPE (SMDS).
4. S'il existe des messages déchargés pour une structure d'application, attendez que ces messages aient été extraits de la file d'attente. Le nombre de messages déchargés signalés par le fichier SMDS (DISPLAY USAGE TYPE) doit être égal à zéro avant de continuer.
5. Arrêtez correctement le gestionnaire de files d'attente à l'aide de la commande STOP QMGR MODE (QUIESCE).
6. Attendez un intervalle au moins équivalent à la valeur du paramètre EXCLINT que vous indiquerez dans la commande BACKUP CFSTRUCT à l'étape suivante.
7. Sur un autre gestionnaire de files d'attente, exécutez une sauvegarde de structure d'unité de couplage pour chaque structure d'unité de couplage récupérable à l'aide de la commande MQSC BACKUP CFSTRUCT et en spécifiant une valeur EXCLINT comme requis à l'étape précédente.
8. Vérifiez que les journaux du gestionnaire de files d'attente ne sont pas nécessaires pour restaurer des structures d'unité de couplage, en examinant la sortie de la commande DISPLAY CFSTATUS (*) TYPE (BACKUP).
9. Utilisez la fonction REMOVE QMGR de l'utilitaire CSQ5PQSG pour supprimer le gestionnaire de files d'attente du groupe de partage de files d'attente. Ce programme est décrit dans [Utilitaire de groupe de partage de files d'attente](#). Un exemple est fourni dans thlqual.SCSQPROC(CSQ45RQM).
10. Avant de redémarrer le gestionnaire de files d'attente, réinitialisez la valeur par défaut du paramètre système QSGDATA et recréez le module de paramètres système. Pour plus d'informations sur la façon de personnaliser vos paramètres système, voir [Utilisation de CSQ6SYSP](#) .

Notez que lorsque vous supprimez le dernier gestionnaire de files d'attente d'un groupe de partage de files d'attente, vous devez utiliser l'option FORCE au lieu de REMOVE. Cela supprime le gestionnaire de files d'attente du groupe de partage de files d'attente, tout en n'effectuant pas les contrôles de cohérence des journaux du gestionnaire de files d'attente requis pour la reprise. Vous ne devez effectuer cette opération que si vous supprimez le groupe de partage de files d'attente.

Suppression d'un groupe de partage de files d'attente des tables Db2

Pour supprimer un groupe de partage de files d'attente des tables Db2, utilisez la fonction REMOVE QSG de l'utilitaire de groupe de partage de files d'attente (CSQ5PQSG). Ce programme est décrit dans [Utilitaire de groupe de partage de files d'attente](#). Un exemple est fourni dans thlqual.SCSQPROC(CSQ45RQS).

Vous ne pouvez supprimer un groupe de partage de files d'attente des tables de groupes de partage de données Db2 communes qu'après avoir supprimé tous les gestionnaires de files d'attente du groupe de partage de files d'attente (comme décrit dans «[Suppression d'un gestionnaire de files d'attente d'un groupe de partage de files d'attente](#)», à la page 415).

Lorsque l'enregistrement de groupe de partage de files d'attente est supprimé de la table d'administration de groupe de partage de files d'attente, tous les objets et les informations d'administration relatifs à ce groupe de partage de files d'attente sont supprimés des autres tables IBM MQ Db2. Cela inclut les informations d'objet de groupe et de file d'attente partagée.

Validation de la cohérence des définitions Db2

Des problèmes peuvent se produire pour les files d'attente partagées au sein d'un groupe de partage de files d'attente si les définitions d'objet Db2 sont devenues incohérentes pour une raison quelconque.

Pour valider la cohérence des définitions d'objet Db2 pour les gestionnaires de files d'attente, les structures d'unité de couplage et les files d'attente partagées, utilisez la fonction VERIFY QSG de l'utilitaire de partage de files d'attente (CSQ5PQSG). Ce programme est décrit dans [Utilitaire de groupe de partage de files d'attente](#).

Gestion des files d'attente partagées

Cette rubrique explique comment récupérer, déplacer et migrer des files d'attente partagées.

Cette section décrit les tâches suivantes:

- «[Récupération des files d'attente partagées](#)», à la page 416
- «[Déplacement de files d'attente partagées](#)», à la page 417
- «[Migration de files d'attente non partagées vers des files d'attente partagées](#)», à la page 420
- [Suspension d'une connexion Db2](#)

Récupération des files d'attente partagées

IBM MQ peut récupérer des messages persistants dans des files d'attente partagées si tous:

- Des sauvegardes des structures CF contenant les messages ont été effectuées.
- Tous les journaux de tous les gestionnaires de files d'attente du groupe de partage de files d'attente sont disponibles pour effectuer une récupération à partir du point où les sauvegardes sont effectuées.
- Db2 est disponible et la table de sauvegarde de structure est plus récente que la sauvegarde de structure d'unité de couplage la plus récente.

Les messages d'une file d'attente partagée sont stockés dans une structure d'unité de couplage. Les messages persistants peuvent être placés dans des files d'attente partagées et, comme les messages persistants des files d'attente non partagées, ils sont copiés dans le journal du gestionnaire de files d'attente. Les commandes MQSC BACKUP CFSTRUCT et RECOVER CFSTRUCT sont fournies pour permettre la récupération d'une structure d'unité de couplage en cas de défaillance improbable d'une unité de couplage. Dans ce cas, tous les messages non persistants stockés dans la structure affectée sont perdus, mais les messages persistants peuvent être récupérés. Toute autre activité d'application utilisant la structure est empêchée tant que la structure n'a pas été récupérée.

Pour activer la reprise, vous devez sauvegarder fréquemment vos structures de liste d'unités de couplage à l'aide de la commande MQSC BACKUP CFSTRUCT. Les messages de la structure d'unité de couplage sont écrits dans le fichier journal actif du gestionnaire de files d'attente effectuant la sauvegarde. Il écrit

un enregistrement de la sauvegarde dans Db2: le nom de la structure d'unité de couplage en cours de sauvegarde, le nom du gestionnaire de files d'attente effectuant la sauvegarde, la plage d'adresses relatives en octets de cette sauvegarde sur ce journal de gestionnaire de files d'attente et l'heure de la sauvegarde. Sauvegardez les structures de liste d'unités de couplage même si vous n'utilisez pas activement les files d'attente partagées, par exemple, si vous avez configuré un groupe de partage de files d'attente avec l'intention de l'utiliser ultérieurement.

Vous pouvez récupérer une structure d'unité de couplage en émettant une commande MQSC RECOVER CFSTRUCT sur le gestionnaire de files d'attente qui peut effectuer la récupération ; vous pouvez utiliser n'importe quel gestionnaire de files d'attente dans le groupe de partage de files d'attente. Vous pouvez spécifier une seule structure d'unité de couplage à récupérer, ou vous pouvez récupérer plusieurs structures d'unité de couplage simultanément.

Comme indiqué précédemment, il est important que vous sauvegardiez fréquemment vos structures de liste CF, sinon la récupération d'une structure CF peut prendre beaucoup de temps. En outre, le processus de récupération ne peut pas être annulé.

La définition d'une file d'attente partagée est conservée dans une base de données Db2 et peut donc être récupérée si nécessaire à l'aide de procédures de base de données Db2 standard. Pour plus d'informations, voir [Files d'attente partagées et groupes de partage de files d'attente](#) .

Déplacement de files d'attente partagées

Cette section explique comment effectuer l'équilibrage de charge en déplaçant une file d'attente partagée d'une structure d'unité de couplage vers une autre. Il décrit également comment déplacer une file d'attente non partagée vers une file d'attente partagée et comment déplacer une file d'attente partagée vers une file d'attente non partagée.

Lorsque vous déplacez une file d'attente, vous devez définir une file d'attente temporaire dans le cadre de la procédure. En effet, chaque file d'attente doit avoir un nom unique, de sorte que vous ne pouvez pas avoir deux files d'attente du même nom, même si les files d'attente ont des dispositions de file d'attente différentes. IBM MQ tolère d'avoir deux files d'attente avec le même nom (comme à l'étape «2», à la page 417), mais vous ne pouvez pas utiliser les files d'attente.

- Déplacement d'une file d'attente d'une structure d'unité de couplage vers une autre
- Déplacement d'une file d'attente non partagée vers une file d'attente partagée
- Déplacement d'une file d'attente partagée vers une file d'attente non partagée

Déplacement d'une file d'attente d'une structure d'unité de couplage vers une autre

Pour déplacer des files d'attente et leurs messages d'une structure d'unité de couplage à une autre, utilisez la commande MQSC MOVE QLOCAL . Une fois que vous avez identifié la ou les files d'attente que vous souhaitez déplacer vers une nouvelle structure d'unité de couplage, utilisez la procédure suivante pour déplacer chaque file d'attente:

1. Assurez-vous que la file d'attente que vous souhaitez déplacer n'est utilisée par aucune application, c'est-à-dire que les attributs de file d'attente IPPROCS et OPPROCS sont égaux à zéro sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente.
2. Empêchez les applications d'insérer des messages dans la file d'attente en cours de déplacement en modifiant la définition de la file d'attente pour désactiver MQPUT s. Remplacez la définition de file d'attente par PUT (DISABLED).
3. Définissez une file d'attente temporaire avec les mêmes attributs que la file d'attente en cours de déplacement à l'aide de la commande suivante:

```
DEFINE QL(TEMP_QUEUE) LIKE(QUEUE_TO_MOVE) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
```

Remarque : Si cette file d'attente temporaire existe à partir d'une exécution précédente, supprimez-la avant d'effectuer la définition.

4. Déplacez les messages dans la file d'attente temporaire à l'aide de la commande suivante:

```
MOVE QLOCAL(Queue_To_Move) TOQLOCAL(TEMP_Queue)
```

5. Supprimez la file d'attente que vous déplacez à l'aide de la commande suivante:

```
DELETE QLOCAL(Queue_To_Move)
```

6. Redéfinissez la file d'attente en cours de déplacement en modifiant l'attribut CFSTRUCT à l'aide de la commande suivante:

```
DEFINE QL(Queue_To_Move) LIKE(TEMP_Queue) CFSTRUCT(NEW) QSGDISP(SHARED)
```

Lorsque la file d'attente est redéfinie, elle est basée sur la file d'attente temporaire créée à l'étape «3», à la page 417.

7. Remplacez les messages dans la nouvelle file d'attente à l'aide de la commande suivante:

```
MOVE QLOCAL(TEMP) TOQLOCAL(Queue_To_Move)
```

8. La file d'attente créée à l'étape «3», à la page 417 n'est plus requise. Utilisez la commande suivante pour la supprimer:

```
DELETE QL(TEMP_Queue)
```

9. Si la file d'attente en cours de déplacement a été définie dans les fichiers CSQINP2, modifiez l'attribut CFSTRUCT de la commande DEFINE QLOCAL appropriée dans les fichiers CSQINP2. Ajoutez le mot clé REPLACE pour remplacer la définition de file d'attente existante.

La Figure 47, à la page 419 illustre un exemple de travail permettant de déplacer une file d'attente d'une structure d'unité de couplage à une autre.

```

//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqua1.SCSQANLE,DISP=SHR
//      DD DSN=thlqua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*

```

Figure 47. Exemple de travail permettant de déplacer une file d'attente d'une structure d'unité de couplage à une autre

Déplacement d'une file d'attente non partagée vers une file d'attente partagée

La procédure de déplacement d'une file d'attente non partagée vers une file d'attente partagée est similaire à la procédure de déplacement d'une file d'attente d'une structure d'unité de couplage vers une autre (voir «Déplacement d'une file d'attente d'une structure d'unité de couplage vers une autre», à la page 417). Figure 48, à la page 419 fournit un exemple de travail à cet effet.

Remarque : N'oubliez pas que les messages des files d'attente partagées sont soumis à certaines restrictions concernant la taille maximale des messages, la persistance des messages et le type d'index de file d'attente. Vous risquez donc de ne pas pouvoir déplacer certaines files d'attente non partagées vers une file d'attente partagée.

```

//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=thlqua1.SCSQANLE,DISP=SHR
//      DD DSN=thlqua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*

```

Figure 48. Exemple de travail de déplacement d'une file d'attente non partagée vers une file d'attente partagée

Déplacement d'une file d'attente partagée vers une file d'attente non partagée

La procédure de déplacement d'une file d'attente partagée vers une file d'attente non partagée est similaire à la procédure de déplacement d'une file d'attente d'une structure d'unité de couplage vers une autre (voir «Déplacement d'une file d'attente d'une structure d'unité de couplage vers une autre», à la page 417).

Figure 49, à la page 420 fournit un exemple de travail à cet effet.

```

//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=th1qua1.SCSQANLE,DISP=SHR
// DD DSN=th1qua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
ALTER QL(Queue_TO_MOVE) PUT(DISABLED)
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(Queue_TO_MOVE) PUT(ENABLED) GET(ENABLED) QSGDISP(QMGR)
MOVE QLOCAL(Queue_TO_MOVE) TOQLOCAL(TEMP_QUEUE)
DELETE QL(Queue_TO_MOVE)
DEFINE QL(Queue_TO_MOVE) LIKE(TEMP_QUEUE) STGCLASS(NEW) QSGDISP(QMGR)
MOVE QLOCAL(TEMP_QUEUE) TOQLOCAL(Queue_TO_MOVE)
DELETE QL(TEMP_QUEUE)
/*

```

Figure 49. Exemple de travail de déplacement d'une file d'attente partagée vers une file d'attente non partagée

Migration de files d'attente non partagées vers des files d'attente partagées

La migration de files d'attente non partagées vers des files d'attente partagées s'effectue en deux étapes:

- Migration du premier gestionnaire de files d'attente (ou uniquement) dans le groupe de partage de files d'attente
- Migration de tout autre gestionnaire de files d'attente dans le groupe de partage de files d'attente

Migration du premier gestionnaire de files d'attente (ou uniquement) dans le groupe de partage de files d'attente

La Figure 48, à la page 419 illustre un exemple de travail de déplacement d'une file d'attente non partagée vers une file d'attente partagée. Effectuez cette opération pour chaque file d'attente à migrer.

Remarque :

1. Les messages des files d'attente partagées sont soumis à certaines restrictions concernant la taille maximale des messages, la persistance des messages et le type d'index de file d'attente. Il est donc possible que vous ne puissiez pas déplacer certaines files d'attente non partagées vers une file d'attente partagée.
2. Vous devez utiliser le type d'index correct pour les files d'attente partagées. Si vous migrez une file d'attente de transmission pour qu'elle soit une file d'attente partagée, le type d'index doit être MSGID.

Si la file d'attente est vide ou que vous n'avez pas besoin de conserver les messages qu'elle contient, la migration de la file d'attente est plus simple. La Figure 50, à la page 421 montre un exemple de travail à utiliser dans ces circonstances.

```
//UTILITY EXEC PGM=CSQUTIL,PARM=('CSQ1')
//STEPLIB DD DSN=th1qua1.SCSQANLE,DISP=SHR
// DD DSN=th1qua1.SCSQAUTH,DISP=SHR
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
COMMAND DDNAME(MOVEQ)
/*
//MOVEQ DD *
DELETE QL(TEMP_QUEUE) PURGE
DEFINE QL(TEMP_QUEUE) LIKE(QUEUE_TO_MOVE) PUT(ENABLED) GET(ENABLED)
DELETE QL(QUEUE_TO_MOVE)
DEFINE QL(QUEUE_TO_MOVE) LIKE(TEMP_QUEUE) CFSTRUCT(NEW) QSGDISP(SHARED)
DELETE QL(TEMP_QUEUE)
/*
```

Figure 50. Exemple de travail de déplacement d'une file d'attente non partagée sans messages vers une file d'attente partagée

Migration de tout autre gestionnaire de files d'attente dans le groupe de partage de files d'attente

1. Pour chaque file d'attente qui n'a pas le même nom qu'une file d'attente partagée existante, déplacez la file d'attente comme décrit dans [Figure 48](#), à la page 419 ou [Figure 50](#), à la page 421.
2. Pour les files d'attente portant le même nom qu'une file d'attente partagée existante, déplacez les messages vers la file d'attente partagée à l'aide des commandes présentées dans [Figure 51](#), à la page 421.

```
MOVE QLOCAL(QUEUE_TO_MOVE) QSGDISP(QMGR) TOQLOCAL(QUEUE_TO_MOVE)
DELETE QLOCAL(QUEUE_TO_MOVE) QSGDISP(QMGR)
```

Figure 51. Déplacement de messages d'une file d'attente non partagée vers une file d'attente partagée existante

Interruption d'une connexion à Db2

Si vous souhaitez appliquer la maintenance ou le service aux tables ou au package Db2 liés aux files d'attente partagées sans arrêter votre gestionnaire de files d'attente, vous devez déconnecter temporairement les gestionnaires de files d'attente du groupe de partage de données (DSG) d' Db2.

Pour ce faire :

1. Utilisez la commande MQSC SUSPEND QMGR FACILITY (Db2).
2. Effectuez les liaisons.
3. Reconnectez-vous à Db2 à l'aide de la commande MQSC RESUME QMGR FACILITY (Db2)

Notez que l'utilisation de ces commandes est soumise à des restrictions.



Avvertissement : Lorsque la connexion Db2 est interrompue, les opérations suivantes ne sont pas disponibles. Par conséquent, vous devez effectuer ce travail pendant une période où votre entreprise est au moins occupée.

- Accès aux objets de file d'attente partagée pour l'administration (définition, suppression, modification)
- Démarrage des canaux partagés
- Stockage des messages dans Db2
- Sauvegarde ou restauration de CFSTRUCT

Gestion des objets de groupe

Cette rubrique explique comment utiliser des objets de groupe.

IBM MQ copie automatiquement la définition d'un objet de groupe dans l'ensemble de pages zéro de chaque gestionnaire de files d'attente qui l'utilise. Vous pouvez modifier temporairement la copie de la définition et IBM MQ vous permet d'actualiser les copies de l'ensemble de pages à partir de la copie du référentiel. IBM MQ tente toujours d'actualiser les copies de l'ensemble de pages à partir de la copie du référentiel au démarrage (pour les objets canal, cette opération est effectuée lorsque l'initiateur de canal redémarre). Cela garantit que les copies de l'ensemble de pages reflètent la version sur le référentiel, y compris les modifications apportées lorsque le gestionnaire de files d'attente était inactif.

Dans certains cas, l'actualisation n'est pas effectuée, par exemple:

- Si une copie de la file d'attente est ouverte, une actualisation qui modifierait l'utilisation de la file d'attente échoue.
- Si une copie d'une file d'attente contient des messages, une actualisation qui supprimerait cette file d'attente échoue.

Dans ces circonstances, l'actualisation n'est pas effectuée sur cette copie, mais sur les copies de tous les autres gestionnaires de files d'attente. Recherchez et corrigez les problèmes liés aux objets de copie après l'ajout, la modification ou la suppression d'un objet de groupe et au redémarrage du gestionnaire de files d'attente ou de l'initiateur de canal.

Gestion de l'unité de couplage

Cette rubrique explique comment ajouter ou supprimer des structures d'unité de couplage (CF).

Cette section décrit les tâches suivantes:

- [«Ajout d'une structure d'unité de couplage»](#), à la page 422
- [«Suppression d'une structure d'unité de couplage»](#), à la page 422

Ajout d'une structure d'unité de couplage

Pour ajouter une structure d'unité de couplage, procédez comme suit:

1. Définissez la structure d'unité de couplage dans le fichier de règles CFRM. Les informations relatives à la configuration de l'unité de couplage dans la rubrique [Configuration de l'unité de couplage](#) décrivent les règles de dénomination des structures d'unité de couplage et expliquent comment définir des structures dans le fichier de règles CFRM.
2. Si vous souhaitez configurer la structure pour décharger les données de message dans le fichier SMDS, allouez et préformatez les fichiers. Pour plus d'informations, voir [Création d'un fichier de messages partagé](#).
3. Définissez la structure dans IBM MQ à l'aide de la commande [DEFINE CFSTRUCT](#).

Suppression d'une structure d'unité de couplage

Pour supprimer une structure d'unité de couplage, procédez comme suit:

1. Utilisez la commande suivante pour obtenir la liste de toutes les files d'attente à l'aide de la structure d'unité de couplage à supprimer:

```
DISPLAY QUEUE(*) QSGDISP(SHARED) CFSTRUCT(structure-name)
```

2. Supprimez toutes les files d'attente qui utilisent la structure.

3. Supprimez la structure d'unité de couplage de IBM MQ à l'aide de la commande `DELETE CFSTRUCT` .
4. Si la structure a été configurée pour décharger les données de message dans le fichier SMDS, supprimez le fichier SMDS.
5. Supprimez la définition de structure de votre fichier de règles CFRM et exécutez l'utilitaire IXCMIAPU. (C'est l'inverse de la tâche de personnalisation qui a configuré l'unité de couplage, décrite dans [Configuration de l'unité de couplage](#).)

Optimisation de la surveillance de la liste des unités de couplage

Utilisez cette rubrique pour comprendre la surveillance de la liste des unités de couplage

La surveillance de liste d'unité de couplage (CF) permet de surveiller l'état des structures de liste contenant des files d'attente partagées IBM MQ . Lorsqu'un message est ajouté à une file d'attente partagée et que la longueur de la file d'attente passe de zéro à une valeur différente de zéro, l'unité de couplage en informe tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Lorsqu'ils sont notifiés, les gestionnaires de files d'attente peuvent effectuer un certain nombre d'actions, notamment la notification des moniteurs de déclenchement qui utilisent TRIGGER (FIRST) ou des applications qui exécutent une opération get-wait.

Par défaut, l'unité de couplage avertit simultanément tous les gestionnaires de files d'attente du groupe de partage de files d'attente. Dans certaines configurations, cela peut entraîner des problèmes, tels que:

- Distribution de la charge de travail asymétrique, où un pourcentage important de messages est acheminé vers un gestionnaire de files d'attente particulier du groupe de partage de files d'attente, souvent le gestionnaire de files d'attente qui s'exécute sur la partition logique la plus rapide ou qui est le plus proche de l'unité de couplage, ou
- Un grand nombre d'extractions ayant échoué, entraînant une perte de temps UC.

z/OS V2R3 introduit un nouvel attribut de gestionnaire de ressources d'unité de couplage (CFRM) appelé **KEYRNOTIFYDELAY**, qui peut être utilisé avec des structures de liste contenant des files d'attente partagées (c'est-à-dire des structures d'application et non la structure d'administration) et qui peut, pour certaines charges de travail, réduire les effets de l'asymétrie de la charge de travail et des appels MQGET vides ou des appels MQGET vides.

KEYRNOTIFYDELAY ne peut être défini que sur les structures d'une fonction CF, s'exécutant au niveau CFLEVEL 22 ou supérieur.

Sa valeur doit être comprise entre un et sept chiffres décimaux, compris entre 0 et 1 000 000 microsecondes. Si cette valeur est définie sur une valeur différente de zéro et que la longueur d'une file d'attente passe de zéro à une valeur différente de zéro, l'unité de couplage sélectionne un gestionnaire de files d'attente unique dans le groupe de partage de files d'attente et notifie ce gestionnaire de files d'attente avant tous les autres gestionnaires de files d'attente du groupe.

Le gestionnaire de files d'attente est sélectionné de manière circulaire. Si le gestionnaire de files d'attente sélectionné ne traite pas le message dans l'intervalle de temps décrit par **KEYRNOTIFYDELAY** , tous les autres gestionnaires de files d'attente du groupe de partage de files d'attente seront également notifiés.

Pour plus d'informations sur **KEYRNOTIFYDELAY** , voir: [Understanding Keyrange Monitoring Notification Delay](#).

Notez qu'il existe deux attributs CFRM similaires appelés **LISTNOTIFYDELAY** et **SUBNOTIFYDELAY**. Aucun de ces éléments n'a d'effet mesurable sur la charge de travail IBM MQ .

Reprise et redémarrage sur z/OS

Utilisez cette rubrique pour comprendre les mécanismes de reprise et de redémarrage utilisés par IBM MQ.

Redémarrage de IBM MQ

Après l'arrêt d'un gestionnaire de files d'attente, des procédures de redémarrage différentes sont nécessaires en fonction de la manière dont le gestionnaire de files d'attente s'est arrêté. Utilisez cette rubrique pour comprendre les différentes procédures de redémarrage que vous pouvez utiliser.

Cette rubrique contient des informations sur le redémarrage de votre gestionnaire de files d'attente dans les cas suivants:

- [«Redémarrage après un arrêt normal», à la page 424](#)
- [«Redémarrage après un arrêt anormal», à la page 424](#)
- [«Redémarrage si vous avez perdu vos ensembles de pages», à la page 424](#)
- [«Redémarrage si vous avez perdu vos fichiers journaux», à la page 424](#)
- [Redémarrage si vous avez perdu vos structures d'unité de couplage](#)

Redémarrage après un arrêt normal

Si le gestionnaire de files d'attente a été arrêté à l'aide de la commande STOP QMGR, le système termine son travail de manière ordonnée et prend un point de contrôle d'arrêt avant de s'arrêter. Lorsque vous redémarrez le gestionnaire de files d'attente, il utilise les informations du point de contrôle du système et du journal de reprise pour déterminer le statut du système lors de l'arrêt.

Pour redémarrer le gestionnaire de files d'attente, émettez la commande START QMGR comme décrit dans [«Démarrage et arrêt d'un gestionnaire de files d'attente sous z/OS», à la page 348](#).

Redémarrage après un arrêt anormal

IBM MQ détecte automatiquement si le redémarrage suit un arrêt normal ou un arrêt anormal.

Le démarrage du gestionnaire de files d'attente après son arrêt anormal est différent de son démarrage après l'émission de la commande STOP QMGR. Si le gestionnaire de files d'attente s'arrête de manière anormale, il s'arrête sans pouvoir terminer son travail ou prendre un point de contrôle d'arrêt.

Pour redémarrer le gestionnaire de files d'attente, émettez la commande START QMGR comme décrit dans [«Démarrage et arrêt d'un gestionnaire de files d'attente sous z/OS», à la page 348](#). Lorsque vous redémarrez un gestionnaire de files d'attente après un arrêt anormal, il actualise sa connaissance de son statut lors de l'arrêt à l'aide des informations du journal et vous informe du statut des différentes tâches.

Normalement, le processus de redémarrage résout tous les états incohérents. Mais, dans certains cas, vous devez prendre des mesures spécifiques pour résoudre les incohérences. Ceci est décrit dans [«Récupération manuelle des unités de travail», à la page 439](#).

Redémarrage si vous avez perdu vos ensembles de pages

Si vous avez perdu vos ensembles de pages, vous devez les restaurer à partir de vos copies de sauvegarde avant de pouvoir redémarrer le gestionnaire de files d'attente. Ceci est décrit dans [«Comment sauvegarder et récupérer des ensembles de pages», à la page 406](#).

Le redémarrage du gestionnaire de files d'attente peut prendre beaucoup de temps dans ces circonstances en raison du temps nécessaire à la reprise sur incident lié au support.

Redémarrage si vous avez perdu vos fichiers journaux

Si, après l'arrêt d'un gestionnaire de files d'attente (à l'aide de la commande STOP QMGR), les deux copies du journal sont perdues ou endommagées, vous pouvez redémarrer le gestionnaire de files

d'attente à condition de disposer d'un ensemble cohérent d'ensembles de pages (généré à l'aide de la [Méthode 1: Sauvegarde intégrale](#)).

Suivez la procédure ci-dessous :

1. Définissez de nouveaux ensembles de pages pour qu'ils correspondent à chaque ensemble de pages existant dans votre gestionnaire de files d'attente. Pour plus d'informations sur la définition d'ensemble de pages, voir [Tâche 15: Définir vos ensembles de pages](#) .
Assurez-vous que chaque nouvel ensemble de pages est plus grand que l'ensemble de pages source correspondant.
2. Utilisez la fonction FORMAT de CSQUTIL pour formater l'ensemble de pages de destination. Voir [Mise en forme des ensembles de pages](#) pour plus de détails.
3. Utilisez la fonction RESETPAGE de CSQUTIL pour copier les ensembles de pages existants ou les réinitialiser en place, puis réinitialisez l'adresse relative en octets du journal dans chaque page. Pour plus d'informations sur cette fonction, voir [Copie d'un ensemble de pages et réinitialisation du journal](#) .
4. Redéfinissez les fichiers journaux du gestionnaire de files d'attente et le fichier d'amorce à l'aide de la commande CSQJU003 (voir [L'utilitaire de modification de l'inventaire des journaux](#)).
5. Redémarrez le gestionnaire de files d'attente à l'aide des nouveaux ensembles de pages. Pour ce faire, effectuez l'une des opérations suivantes:
 - Modifiez la procédure de tâche démarrée du gestionnaire de files d'attente pour référencer les nouveaux ensembles de pages. Pour plus d'informations, voir [Tâche 6: Créer des procédures pour le gestionnaire de files d'attente IBM MQ](#) .
 - Utilisez les services de méthode d'accès pour supprimer les anciens ensembles de pages, puis renommez les nouveaux ensembles de pages, en leur attribuant les mêmes noms que les anciens ensembles de pages.

Avertissement : Avant de supprimer un ensemble de pages IBM MQ , vérifiez que vous avez effectué les copies de sauvegarde requises.

Si le gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente, les définitions d'objet GROUP et SHARED ne sont normalement pas affectées par des journaux perdus ou endommagés. Toutefois, si des messages de file d'attente partagée sont impliqués dans une unité d'oeuvre couverte par les journaux perdus ou endommagés, l'effet sur ces messages non validés est imprévisible.

Remarque : Si les journaux sont endommagés et que le gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente, la possibilité de récupérer des messages persistants partagés peut être perdue. Emettez immédiatement une commande BACKUP CFSTRUCT sur un autre gestionnaire de files d'attente actif dans le groupe de partage de files d'attente pour toutes les structures d'unité de couplage avec l'attribut RECOVER (YES).

Redémarrage si vous avez perdu vos structures CF

Vous n'avez pas besoin de redémarrer si vous perdez vos structures d'unité de couplage, car le gestionnaire de files d'attente ne s'arrête pas.

Reprise sur site alternative sous z/OS

Vous pouvez récupérer un gestionnaire de files d'attente unique ou un groupe de partage de files d'attente, ou envisager la mise en miroir du disque.

Pour plus de détails, voir les sections suivantes:

- [Récupération d'un gestionnaire de files d'attente unique sur un autre site](#)
- [Récupération d'un groupe de partage de files d'attente.](#)
 - [Récupération du support de structure d'unité de couplage](#)
 - [Sauvegarde du groupe de partage de files d'attente sur le site principal](#)
 - [Récupération d'un groupe de partage de files d'attente sur le site alternatif](#)

- Utilisation de la mise en miroir du disque

Récupération d'un seul gestionnaire de files d'attente sur un autre site

Si une perte totale d'un centre de calcul IBM MQ se produit, vous pouvez effectuer une reprise sur un autre gestionnaire de files d'attente ou groupe de partage de files d'attente sur un site de reprise. (Voir [«Récupération d'un groupe de partage de files d'attente sur le site de remplacement»](#), à la page 430 pour la procédure de reprise sur site alternative pour un groupe de partage de files d'attente.)

Pour effectuer une reprise sur un autre gestionnaire de files d'attente sur un site de reprise, vous devez régulièrement sauvegarder les ensembles de pages et les journaux. Comme pour toutes les opérations de reprise des données, les objectifs de la reprise après incident sont de perdre le moins de données, le traitement de la charge de travail (mises à jour) et le temps, autant que possible.

Sur le site de récupération:

- Les gestionnaires de files d'attente de reprise **doivent** avoir les mêmes noms que les gestionnaires de files d'attente perdus.
- Le module de paramètres système (par exemple, CSQZPARM) utilisé sur chaque gestionnaire de files d'attente de reprise doit contenir les mêmes paramètres que le gestionnaire de files d'attente perdu correspondant.

Une fois cette opération effectuée, rétablissez tous vos gestionnaires de files d'attente, comme décrit dans la procédure suivante. Cela peut être utilisé pour effectuer une reprise après incident sur le site de reprise pour un seul gestionnaire de files d'attente. Il suppose que tout ce qui est disponible est:

- Copies des journaux d'archivage et des fichiers d'amorce créés par une exécution normale sur le site principal (les journaux actifs auront été perdus avec le gestionnaire de files d'attente sur le site principal).
- Les copies des ensembles de pages du gestionnaire de files d'attente sur le site principal qui ont la même ancienneté ou la même ancienneté que les copies de journaux d'archivage les plus récentes disponibles.

Vous pouvez utiliser la double consignation pour les journaux actifs et d'archivage, auquel cas vous devez appliquer les mises à jour du fichier d'amorce aux deux copies:

1. Définissez de nouveaux ensembles de données d'ensemble de pages et chargez-les avec les données des copies des ensembles de pages à partir du site principal.
2. Définissez de nouveaux fichiers journaux actifs.
3. Définissez un nouveau fichier d'amorce et utilisez Access Method Services REPRO pour y copier le fichier d'amorce archivé *le plus récent* .
4. Utilisez l'utilitaire d'impression de mappe de journal CSQJU004 pour imprimer les informations de ce fichier d'amorce le plus récent. Au moment de l'archivage de ce fichier d'amorce, le journal archivé le plus récent que vous avez vient d'être tronqué en tant que journal actif et n'apparaît pas en tant que journal archivé. Enregistrez les adresses STARTRBA et ENDRBA de ce journal.
5. Utilisez l'utilitaire d'inventaire des journaux de modifications, CSQJU003, pour enregistrer ce dernier fichier d'archivage dans le fichier d'amorce que vous venez de restaurer, à l'aide des paramètres STARTRBA et ENDRBA enregistrés à l'étape [«4»](#), à la page 426.
6. Utilisez l'option DELETE de CSQJU003 pour supprimer toutes les informations de journal actives du fichier d'amorce.
7. Utilisez l'option NEWLOG de CSQJU003 pour ajouter des journaux actifs au fichier d'amorce, ne spécifiez pas STARTRBA ou ENDRBA.
8. Utilisez CSQJU003 pour ajouter un enregistrement de contrôle de redémarrage au fichier d'amorce. Indiquez CRESTART CREATE, ENDRBA=highrba, où highrba est l'adresse relative en octets la plus élevée du journal d'archivage le plus récent disponible (à l'étape [«4»](#), à la page 426), plus 1.

Le fichier d'amorce décrit désormais tous les journaux actifs comme étant vides, tous les journaux archivés dont vous disposez et aucun point de contrôle au-delà de la fin de vos journaux.

9. Redémarrez le gestionnaire de files d'attente à l'aide de la commande START QMGR. Lors de l'initialisation, un message de réponse de l'opérateur tel que le suivant est émis:

```
CSQJ245D +CSQ1 RESTART CONTROL INDICATES TRUNCATION AT RBA highrba.  
REPLY Y TO CONTINUE, N TO CANCEL
```

Entrez Y pour démarrer le gestionnaire de files d'attente. Le gestionnaire de files d'attente démarre et récupère les données jusqu'à l'adresse ENDRBA spécifiée dans l'instruction CRESTART.

Pour plus d'informations sur l'utilisation de CSQJU003 et CSQJU004, voir [Utilisation des utilitaires IBM MQ](#).

L'exemple suivant illustre des exemples d'instructions d'entrée pour CSQJU003 pour les étapes 6, 7 et 8:

```
* Step 6  
DELETE DSNAMES=MQM2.LOGCOPY1.DS01  
DELETE DSNAMES=MQM2.LOGCOPY1.DS02  
DELETE DSNAMES=MQM2.LOGCOPY1.DS03  
DELETE DSNAMES=MQM2.LOGCOPY1.DS04  
DELETE DSNAMES=MQM2.LOGCOPY2.DS01  
DELETE DSNAMES=MQM2.LOGCOPY2.DS02  
DELETE DSNAMES=MQM2.LOGCOPY2.DS03  
DELETE DSNAMES=MQM2.LOGCOPY2.DS04  
  
* Step 7  
NEWLOG DSNAMES=MQM2.LOGCOPY1.DS01,COPY1  
NEWLOG DSNAMES=MQM2.LOGCOPY1.DS02,COPY1  
NEWLOG DSNAMES=MQM2.LOGCOPY1.DS03,COPY1  
NEWLOG DSNAMES=MQM2.LOGCOPY1.DS04,COPY1  
NEWLOG DSNAMES=MQM2.LOGCOPY2.DS01,COPY2  
NEWLOG DSNAMES=MQM2.LOGCOPY2.DS02,COPY2  
NEWLOG DSNAMES=MQM2.LOGCOPY2.DS03,COPY2  
NEWLOG DSNAMES=MQM2.LOGCOPY2.DS04,COPY2  
  
* Step 8  
CRESTART CREATE,ENDRBA=063000
```

Les éléments à prendre en compte pour le redémarrage de l'initiateur de canal sur le site de récupération sont similaires à ceux rencontrés lors de l'utilisation d'ARM pour redémarrer l'initiateur de canal sur une image z/OS différente. Pour plus d'informations, voir «Utilisation d'ARM dans un réseau IBM MQ», à la page 436. Votre stratégie de reprise doit également couvrir la reprise des bibliothèques de produits IBM MQ et des environnements de programmation d'application qui utilisent IBM MQ (CICS, par exemple).

D'autres fonctions de l'utilitaire d'inventaire des journaux de modifications (CSQJU003) peuvent également être utilisées dans les scénarios de reprise après incident. La fonction HIGHRBA permet la mise à jour des valeurs les plus élevées de l'adresse relative en octets écrite et de l'adresse relative en octets la plus élevée déchargée dans le fichier d'amorçage. La fonction CHECKPT permet d'ajouter de nouveaux enregistrements de file d'attente de points de contrôle ou de supprimer des enregistrements de file d'attente de points de contrôle existants dans le fichier d'amorce.

Avertissement : Ces fonctions peuvent affecter l'intégrité de vos données IBM MQ . Utilisez-les uniquement dans les scénarios de reprise après incident sous la supervision du personnel de maintenance IBM .

Techniques de copie rapide

Si des copies de tous les ensembles de pages et journaux sont effectuées alors que le gestionnaire de files d'attente est gelé, les copies seront un ensemble cohérent qui peut être utilisé pour redémarrer le gestionnaire de files d'attente sur un autre site. Ils permettent généralement un redémarrage beaucoup plus rapide du gestionnaire de files d'attente, car il y a peu de reprise sur support à effectuer.

Utilisez la commande SUSPEND QMGR LOG pour geler le gestionnaire de files d'attente. Cette commande vide les pools de mémoire tampon dans les ensembles de pages, prend un point de contrôle et arrête toute autre activité d'écriture de journal. Une fois l'activité d'écriture de journal interrompue, le gestionnaire de files d'attente est effectivement gelé jusqu'à ce que vous émettiez

une commande RESUME QMGR LOG. Lorsque le gestionnaire de files d'attente est gelé, les ensembles de pages et les journaux peuvent être copiés.

En utilisant des outils de copie tels que FLASHCOPY ou SNAPSHOT pour copier rapidement les ensembles de pages et les journaux, le temps pendant lequel le gestionnaire de files d'attente est gelé peut être réduit à un minimum.

Au sein d'un groupe de partage de files d'attente, cependant, la commande SUSPEND QMGR LOG peut ne pas être une bonne solution. Pour être efficaces, les copies des journaux doivent toutes contenir le même point de cohérence pour la récupération, ce qui signifie que la commande SUSPEND QMGR LOG doit être émise sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente simultanément et que, par conséquent, l'ensemble du groupe de partage de files d'attente sera gelé pendant un certain temps.

Récupération d'un groupe de partage de files d'attente

En cas de sinistre sur un site principal, vous pouvez redémarrer un groupe de partage de files d'attente sur un site distant à l'aide de jeux de données de sauvegarde à partir du site principal. Pour récupérer un groupe de partage de files d'attente, vous devez coordonner la récupération sur tous les gestionnaires de files d'attente du groupe de partage de files d'attente et la coordonner avec d'autres ressources, principalement Db2. Cette section décrit ces tâches en détail.

- [Récupération du support de structure d'unité de couplage](#)
- [Sauvegarde du groupe de partage de files d'attente sur le site principal](#)
- [Récupération d'un groupe de partage de files d'attente sur le site alternatif](#)

Reprise sur support de structure d'unité de couplage

La reprise sur incident lié au support d'une structure d'unité de couplage utilisée pour stocker les messages persistants dans une file d'attente partagée repose sur une sauvegarde du support qui peut être récupérée en aval par l'application des mises à jour consignées. Effectuez des sauvegardes périodiques de vos structures CF à l'aide de la commande MQSC BACKUP CFSTRUCT. Toutes les mises à jour des files d'attente partagées (MQGETs et MQPUTs) sont écrites dans le journal du gestionnaire de files d'attente où la mise à jour est effectuée. Pour effectuer une reprise sur incident lié au support d'une structure d'unité de couplage, vous devez appliquer les mises à jour consignées à cette sauvegarde à partir des journaux de tous les gestionnaires de files d'attente qui ont utilisé cette structure d'unité de couplage. Lorsque vous utilisez la commande MQSC RECOVER CFSTRUCT, IBM MQ fusionne automatiquement les journaux des gestionnaires de files d'attente concernés et applique les mises à jour à la sauvegarde la plus récente.

La sauvegarde de structure d'unité de couplage est écrite dans le journal du gestionnaire de files d'attente qui a traité la commande BACKUP CFSTRUCT, de sorte qu'il n'y a pas de fichiers supplémentaires à collecter et à transporter vers le site de remplacement.

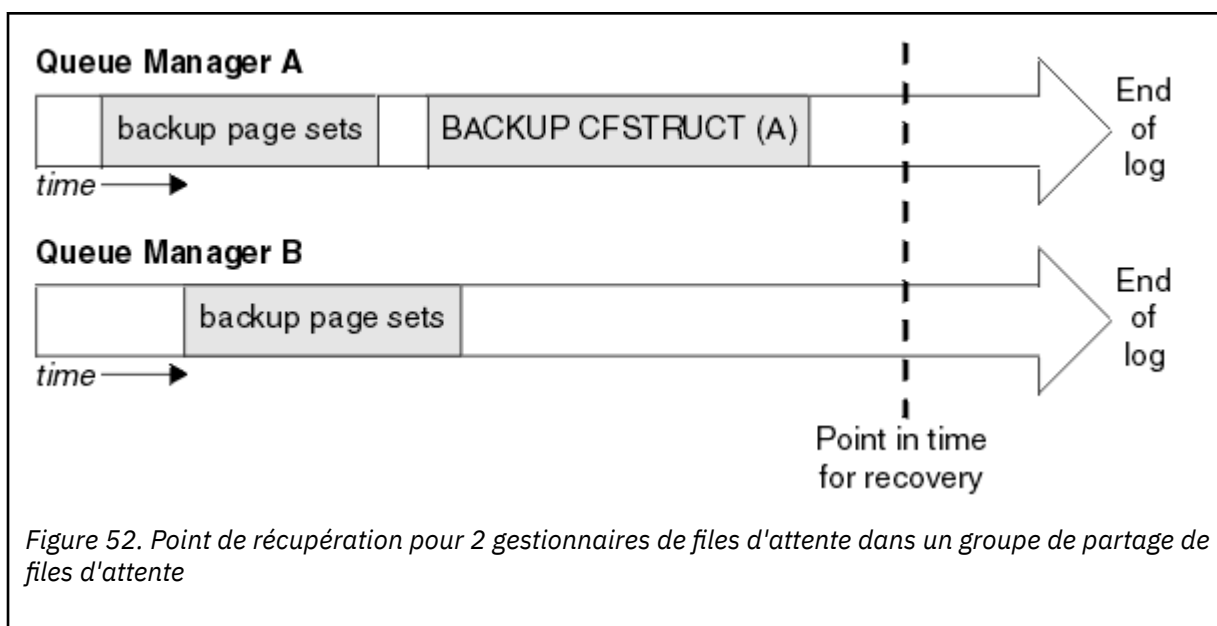
Sauvegarde du groupe de partage de files d'attente sur le site principal

Sur le site principal, vous devez établir un ensemble cohérent de sauvegardes sur une base régulière, qui peut être utilisé en cas de sinistre pour régénérer le groupe de partage de files d'attente sur un autre site. Pour un seul gestionnaire de files d'attente, la récupération peut se faire à un point de cohérence arbitraire, généralement la fin des journaux disponibles sur le site distant. Toutefois, lorsque des messages persistants ont été stockés dans une file d'attente partagée, les journaux de tous les gestionnaires de files d'attente du groupe de partage de files d'attente doivent être fusionnés pour récupérer les files d'attente partagées, car tout gestionnaire de files d'attente du groupe de partage de files d'attente peut avoir effectué des mises à jour (MQPUT s ou MQGET s) sur la file d'attente.

Pour la récupération d'un groupe de partage de files d'attente, vous devez établir un point de cohérence compris dans la plage de journalisation des données de journal de tous les

gestionnaires de files d'attente. Toutefois, comme vous pouvez uniquement **réacheminer** le support de récupération à partir du journal, ce point de cohérence doit être postérieur à l'exécution de la commande BACKUP CFSTRUCT et à l'exécution des sauvegardes d'ensemble de pages. (En règle générale, le moment de la récupération peut correspondre à la fin d'un jour ouvrable ou d'une semaine.)

Le diagramme suivant présente les lignes de temps pour deux gestionnaires de files d'attente d'un groupe de partage de files d'attente. Pour chaque gestionnaire de files d'attente, des sauvegardes partielles des ensembles de pages sont effectuées (voir [Méthode 2: Sauvegarde partielle](#)). Sur le gestionnaire de files d'attente A, une commande BACKUP CFSTRUCT est émise. Ensuite, une commande ARCHIVE LOG est émise sur chaque gestionnaire de files d'attente pour tronquer le journal actif et le copier sur le support hors ligne à partir du gestionnaire de files d'attente, qui peut être transporté vers le site alternatif. La fin du journal identifie l'heure à laquelle la commande ARCHIVE LOG a été émise et marque donc l'étendue des données de journal généralement disponibles sur le site alternatif. Le point de cohérence pour la récupération doit être compris entre la fin des sauvegardes d'ensemble de pages ou de structure d'unité de couplage et la fin la plus précoce du journal disponible sur le site de remplacement.



IBM MQ enregistre les informations associées aux sauvegardes de structure d'unité de couplage dans une table dans Db2. En fonction de vos besoins, vous souhaitez peut-être coordonner le point de cohérence pour la récupération de IBM MQ avec celui de Db2, ou il peut être suffisant de prendre une copie du CSQ IBM MQ CSQ.ADMIN_B_STRBACKUP après l'exécution des commandes BACKUP CFSTRUCT.

Pour préparer une reprise:

1. Créez des sauvegardes d'ensemble de pages pour chaque gestionnaire de files d'attente du groupe de partage de files d'attente.
2. Émettez une commande BACKUP CFSTRUCT pour chaque structure d'unité de couplage avec l'attribut RECOVER (YES). Vous pouvez exécuter ces commandes à partir d'un seul gestionnaire de files d'attente ou à partir de différents gestionnaires de files d'attente au sein du groupe de partage de files d'attente pour équilibrer la charge de travail.
3. Une fois toutes les sauvegardes terminées, émettez une commande ARCHIVE LOG pour changer le journal actif et créer des copies des journaux et des fichiers d'amorce de chaque gestionnaire de files d'attente du groupe de partage de files d'attente.
4. Transportez les sauvegardes d'ensemble de pages, les journaux archivés, le fichier d'amorce archivé de tous les gestionnaires de files d'attente du groupe de partage de files d'attente et les informations de sauvegarde Db2 de votre choix, hors site.

Récupération d'un groupe de partage de files d'attente sur le site de remplacement

Avant de pouvoir récupérer le groupe de partage de files d'attente, vous devez préparer l'environnement:

1. Si vous disposez d'anciennes informations dans votre unité de couplage à partir de démarrages de simulation lors de l'installation du groupe de partage de files d'attente, vous devez d'abord nettoyer les éléments suivants:

Remarque : Si vous ne disposez pas d'anciennes informations dans l'unité de couplage, vous pouvez omettre cette étape.

- a. Entrez la commande z/OS suivante pour afficher les structures d'unité de couplage pour ce groupe de partage de files d'attente:

```
D XCF,STRUCTURE,STRNAME= qsgname
```

- b. Pour toutes les structures commençant par le nom du groupe de partage de files d'attente, utilisez la commande z/OS SETXCF FORCE CONNECTION pour forcer la déconnexion de ces structures:

```
SETXCF FORCE,CONNECTION,STRNAME= strname,CONNAME=ALL
```

- c. Supprimez toutes les structures CF à l'aide de la commande suivante pour chaque structure:

```
SETXCF FORCE,STRUCTURE,STRNAME= strname
```

2. Restaurez les systèmes Db2 et les groupes de partage de données.
3. Récupérez le CSQ CSQ.ADMIN_B_STRBACKUP pour qu'elle contienne des informations sur les sauvegardes de structure les plus récentes effectuées sur le site principal.
Remarque : Il est important que la table STRBACKUP contienne les informations de sauvegarde de structure les plus récentes. Les informations de sauvegarde de structure plus ancienne peuvent nécessiter des fichiers que vous avez supprimés suite aux informations fournies par une commande récente DISPLAY USAGE TYPE (DATASET), ce qui signifie que votre structure CF récupérée ne contient pas d'informations précises.
4. Exécutez la commande ADD QMGR de l'utilitaire CSQ5PQSG pour chaque gestionnaire de files d'attente du groupe de partage de files d'attente. Cette opération restaure l'entrée de groupe XCF pour chaque gestionnaire de files d'attente.

Lorsque vous exécutez l'utilitaire dans ce scénario, les messages suivants sont normaux:

```
CSQU566I Unable to get attributes for admin structure, CF not found  
or not allocated  
CSQU546E Unable to add QMGR queue_manager_name entry,  
already exists in DB2 table CSQ.ADMIN_B_QMGR  
CSQU148I CSQ5PQSG Utility completed, return code=4
```

Pour récupérer les gestionnaires de files d'attente dans le groupe de partage de files d'attente:

1. Définissez de nouveaux ensembles de données d'ensemble de pages et chargez-les avec les données des copies des ensembles de pages à partir du site principal.
2. Définissez de nouveaux fichiers journaux actifs.
3. Définissez un nouveau fichier d'amorce et utilisez Access Method Services REPRO pour y copier le fichier d'amorce archivé *le plus récent*.
4. Utilisez l'utilitaire d'impression de mappe de journal CSQJU004 pour imprimer les informations de ce fichier d'amorce le plus récent. Au moment de l'archivage de ce fichier d'amorce, le journal archivé le plus récent que vous avez vient d'être tronqué en tant que journal actif et n'apparaît pas en tant que journal archivé. Enregistrez les valeurs STARTRBA, STARTLRSN, ENDRBA et ENDLRSN de ce journal.

5. Utilisez l'utilitaire de modification de l'inventaire des journaux, CSQJU003, pour enregistrer ce dernier fichier d'archivage dans le fichier d'amorce que vous venez de restaurer, à l'aide des valeurs enregistrées à l'étape «4», à la page 430.
6. Utilisez l'option DELETE de CSQJU003 pour supprimer toutes les informations de journal actives du fichier d'amorce.
7. Utilisez l'option NEWLOG de CSQJU003 pour ajouter des journaux actifs au fichier d'amorce, ne spécifiez pas STARTRBA ou ENDRBA.
8. Calculez le *recoverylrsn* pour le groupe de partage de files d'attente. *recoverylrsn* est le plus bas des ENDLRSNs de tous les gestionnaires de files d'attente du groupe de partage de files d'attente (comme indiqué à l'étape «4», à la page 430), moins 1. Par exemple, s'il existe deux gestionnaires de files d'attente dans le groupe de partage de files d'attente et que le paramètre ENDLRSN de l'un d'entre eux est B713 3C72 22C5 et de l'autre est B713 3D45 2123, le paramètre *recoverylrsn* est B713 3C72 22C4.
9. Utilisez CSQJU003 pour ajouter un enregistrement de contrôle de redémarrage au fichier d'amorce. Spécifier :

```
CRESTART CREATE, ENDLRSN= recoverylrsn
```

où *recoverylrsn* est la valeur que vous avez enregistrée à l'étape «8», à la page 431.

Le fichier d'amorce décrit désormais tous les journaux actifs comme étant vides, tous les journaux archivés dont vous disposez et aucun point de contrôle au-delà de la fin de vos journaux.

Vous devez ajouter l'enregistrement CRESTART au fichier d'amorce de chaque gestionnaire de files d'attente du groupe de partage de files d'attente.

10. Redémarrez chaque gestionnaire de files d'attente du groupe de partage de files d'attente à l'aide de la commande START QMGR. Lors de l'initialisation, un message de réponse de l'opérateur tel que le suivant est émis:

```
CSQJ245D +CSQ1 RESTART CONTROL INDICATES TRUNCATION AT RBA highrba.  
REPLY Y TO CONTINUE, N TO CANCEL
```

Répondez Y pour démarrer le gestionnaire de files d'attente. Le gestionnaire de files d'attente démarre et récupère les données jusqu'à l'adresse ENDRBA spécifiée dans l'instruction CRESTART.

Pour IBM WebSphere MQ 7.0.1 et les versions ultérieures, le premier gestionnaire de files d'attente démarré peut régénérer les partitions de la structure d'administration pour les autres membres du groupe de partage de files d'attente ainsi que les siennes, et il n'est plus nécessaire de redémarrer chaque gestionnaire de files d'attente du groupe de partage de files d'attente à ce stade.

11. Lorsque les données de la structure d'administration de tous les gestionnaires de files d'attente ont été régénérées, émettez une commande RECOVER CFSTRUCT pour chaque structure d'application CF.

Si vous émettez la commande RECOVER CFSTRUCT pour toutes les structures d'un seul gestionnaire de files d'attente, le processus de fusion des journaux n'est exécuté qu'une seule fois. Il est donc plus rapide que d'émettre la commande sur un gestionnaire de files d'attente différent pour chaque structure d'unité de couplage, où chaque gestionnaire de files d'attente doit effectuer l'étape de fusion des journaux.

Lorsque le traitement du redémarrage conditionnel est utilisé dans un groupe de partage de files d'attente, IBM WebSphere MQ 7.0.1 et les gestionnaires de files d'attente ultérieurs, effectuant une régénération d'administrateur d'homologue, vérifiez que les fichiers d'amorce des homologues contiennent le même CRESTART LRSN que le leur. Cela permet de garantir l'intégrité de la structure d'administration régénérée. Il est donc important de redémarrer d'autres homologues dans le groupe de partage de files d'attente, afin qu'ils puissent traiter leurs propres informations CRESTART, avant le prochain redémarrage inconditionnel d'un membre du groupe.

Utilisation de la mise en miroir du disque

De nombreuses installations utilisent désormais des technologies de mise en miroir de disque telles que IBM Metro Mirror (anciennement PPRC) pour effectuer des copies synchrones des fichiers sur un autre site. Dans de telles situations, la plupart des étapes détaillées deviennent inutiles car les ensembles de pages et les journaux IBM MQ sur le site de remplacement sont effectivement identiques à ceux du site principal. Lorsque de telles technologies sont utilisées, les étapes de redémarrage d'un groupe de partage de files d'attente sur un autre site peuvent être résumées comme suit:

- Désélectionnez les structures d'unité de couplage IBM MQ sur le site de remplacement. (Elles contiennent souvent des informations résiduelles de tout exercice de reprise après incident antérieur).
- Restaurez les systèmes Db2 et toutes les tables de la base de données utilisée par le groupe de partage de files d'attente IBM MQ .
- Redémarrez les gestionnaires de files d'attente. Avant IBM WebSphere MQ 7.0.1, il est nécessaire de redémarrer chaque gestionnaire de files d'attente défini dans le groupe de partage de files d'attente car chaque gestionnaire de files d'attente récupère sa propre partition de la structure d'administration lors du redémarrage du gestionnaire de files d'attente. Une fois que chaque gestionnaire de files d'attente a été redémarré, ceux qui ne se trouvent pas sur leur partition logique d'origine peuvent être arrêtés à nouveau. Pour IBM WebSphere MQ 7.0.1 et les versions ultérieures, le premier gestionnaire de files d'attente a démarré la régénération des partitions de structure d'administration pour les autres membres du groupe de partage de files d'attente ainsi que pour les siens, et il n'est plus nécessaire de redémarrer chaque gestionnaire de files d'attente du groupe de partage de files d'attente.
- Une fois la structure d'administration régénérée, récupérez les structures d'application.

Réinitialisation d'un gestionnaire de files d'attente

Si le gestionnaire de files d'attente s'est arrêté de manière anormale, il se peut que vous ne puissiez pas le redémarrer. Cela peut être dû au fait que vos ensembles de pages ou journaux ont été perdus, tronqués ou endommagés. Si cela s'est produit, vous devrez peut-être réinitialiser le gestionnaire de files d'attente (effectuer un démarrage à froid).

Avertissement

N'effectuez un démarrage à froid que si vous ne pouvez pas redémarrer le gestionnaire de files d'attente d'une autre manière. L'exécution d'un démarrage à froid vous permet de récupérer votre gestionnaire de files d'attente et vos définitions d'objet ; vous ne pourrez **pas** récupérer vos données de message. Vérifiez qu'aucun des autres scénarios de redémarrage décrits dans cette rubrique ne fonctionne pour vous avant d'effectuer cette opération.

Une fois que vous avez redémarré, tous vos objets IBM MQ sont définis et disponibles pour être utilisés, mais il n'y a pas de données de message.

Remarque : Ne réinitialisez pas un gestionnaire de files d'attente lorsqu'il fait partie d'un cluster. Vous devez d'abord supprimer le gestionnaire de files d'attente du cluster (à l'aide des commandes RESET CLUSTER sur les autres gestionnaires de files d'attente du cluster), puis le réinitialiser et enfin le réintroduire dans le cluster en tant que nouveau gestionnaire de files d'attente.

En effet, lors de la réinitialisation, l'identificateur de gestionnaire de files d'attente (QMID) est modifié, de sorte que tout objet de cluster avec l'ancien identificateur de gestionnaire de files d'attente doit être supprimé du cluster.

Pour plus d'informations, voir les sections suivantes:

- [Réinitialisation d'un gestionnaire de files d'attente qui ne fait pas partie d'un groupe de partage de files d'attente](#)
- [Réinitialisation des gestionnaires de files d'attente dans un groupe de partage de files d'attente](#)

Réinitialisation d'un gestionnaire de files d'attente qui ne se trouve pas dans un groupe de partage de files d'attente

Pour réinitialiser un gestionnaire de files d'attente, procédez comme suit:

1. Préparez les instructions de définition d'objet à utiliser lorsque vous redémarrez le gestionnaire de files d'attente. Pour ce faire, procédez comme suit:
 - Si l'ensemble de pages zéro est disponible, utilisez la fonction CSQUTIL SDEFS (voir [Production d'une liste de commandes IBM MQ define](#)). Vous devez obtenir des définitions pour tous les types d'objet (objets d'informations d'authentification, structures d'unité de couplage, canaux, listes de noms, processus, files d'attente et classes de stockage).
 - Si l'ensemble de pages zéro n'est pas disponible, utilisez les définitions de la dernière fois que vous avez sauvegardé vos définitions d'objet.
2. Redéfinissez les fichiers de votre gestionnaire de files d'attente (ne le faites pas tant que vous n'avez pas terminé l'étape «1», à la page 433).

Pour plus d'informations, voir [Création des fichiers d'amorçage et de journalisation](#) et [Définition de vos ensembles de pages](#) .

3. Redémarrez le gestionnaire de files d'attente à l'aide des fichiers journaux, du fichier d'amorce et des ensembles de pages nouvellement définis et initialisés. Utilisez les instructions d'entrée de définition d'objet que vous avez créées à l'étape «1», à la page 433 comme entrée dans le fichier d'entrée d'initialisation CSQINP2 .

Réinitialisation des gestionnaires de files d'attente dans un groupe de partage de files d'attente

Dans un groupe de partage de files d'attente, la réinitialisation d'un gestionnaire de files d'attente est plus complexe. Il peut être nécessaire de réinitialiser un ou plusieurs gestionnaires de files d'attente en raison de problèmes d'ensemble de pages ou de journal, mais il peut également y avoir des problèmes avec Db2 ou l'unité de couplage à gérer. Pour cette raison, il existe un certain nombre d'alternatives:

Démarrage à froid

La réinitialisation de l'ensemble du groupe de partage de files d'attente implique de forcer toutes les structures d'unités de couplage, d'effacer toutes les définitions d'objet du groupe de partage de files d'attente de Db2, de supprimer ou de redéfinir les journaux et le fichier d'amorce et de mettre en forme les ensembles de pages pour tous les gestionnaires de files d'attente du groupe de partage de files d'attente.

Définitions partagées conservées

Supprimez ou redéfinissez les journaux et le fichier d'amorce, formatez les ensembles de pages pour tous les gestionnaires de files d'attente du groupe de partage de files d'attente et forcez toutes les structures d'unités de couplage. Au redémarrage, tous les messages auront été supprimés. Les gestionnaires de files d'attente recréent des objets COPY qui correspondent à des objets GROUP qui existent toujours dans la base de données Db2 . Les files d'attente partagées existent toujours et peuvent être utilisées.

Gestionnaire de files d'attente unique réinitialisé

Supprimez ou redéfinissez les journaux et le fichier d'amorce, ainsi que les ensembles de pages de format pour le gestionnaire de files d'attente unique (tous ses objets et messages privés sont supprimés). Au redémarrage, le gestionnaire de files d'attente recrée des objets COPY qui correspondent à des objets GROUP qui existent toujours dans la base de données Db2 . Les files d'attente partagées existent toujours, tout comme les messages qu'elles contiennent, et peuvent être utilisées.

Récupération à un point de cohérence d'un groupe de partage de files d'attente

Il s'agit du scénario alternatif de reprise après incident sur site.

Les objets partagés sont récupérés au point de cohérence atteint par la récupération Db2 (décrite dans [A Db2 system fails](#)). Chaque gestionnaire de files d'attente peut être récupéré à un point dans le temps à partir des copies de sauvegarde disponibles sur le site alternatif.

Les messages persistants peuvent être utilisés dans des groupes de partage de files d'attente et peuvent être récupérés à l'aide de la commande MQSC RECOVER CFSTRUCT. Notez que cette commande récupère au moment de l'échec. Cependant, il n'y a pas de reprise des messages de file d'attente partagée non persistants ; ils sont perdus sauf si vous avez effectué des copies de sauvegarde indépendamment à l'aide de la fonction COPY du programme utilitaire CSQUTIL.

Il n'est pas nécessaire de tenter de restaurer chaque gestionnaire de files d'attente au même point de cohérence car il n'y a pas d'interdépendances entre les objets locaux des différents gestionnaires de files d'attente (qui sont en cours de récupération), et la resynchronisation du gestionnaire de files d'attente avec Db2 au redémarrage crée ou supprime des objets COPY, selon les besoins, sur un gestionnaire de files d'attente par gestionnaire de files d'attente.

Utilisation de z/OS Automatic Restart Manager (ARM)

Cette rubrique explique comment utiliser ARM pour redémarrer automatiquement vos gestionnaires de files d'attente.

Cette section contient des informations sur les rubriques suivantes:

- [«Qu'est-ce que l'ARM?», à la page 434](#)
- [«Règles ARM», à la page 435](#)
- [«Utilisation d'ARM dans un réseau IBM MQ», à la page 436](#)

Qu'est-ce que l'ARM?

z/OS Automatic Restart Manager (ARM) est une fonction de reprise z/OS qui peut améliorer la disponibilité de vos gestionnaires de files d'attente. Lorsqu'un travail ou une tâche échoue, ou que le système sur lequel il s'exécute échoue, le gestionnaire ARM peut redémarrer le travail ou la tâche sans intervention de l'opérateur.

Si un gestionnaire de files d'attente ou un initiateur de canal échoue, ARM le redémarre dans la même image z/OS. Si z/OS, et donc tout un groupe de sous-systèmes et d'applications associés ont échoué, ARM peut redémarrer automatiquement tous les systèmes défaillants, dans un ordre prédéfini, sur une autre image z/OS du sysplex. Il s'agit d'un *redémarrage intersystème*.

Redémarrez l'initiateur de canal par ARM uniquement dans des circonstances exceptionnelles. Si le gestionnaire de files d'attente est redémarré par ARM, redémarrez l'initiateur de canal à partir du fichier d'initialisation CSQINP2 (voir [«Utilisation d'ARM dans un réseau IBM MQ», à la page 436](#)).

Vous devez utiliser le gestionnaire ARM pour redémarrer un gestionnaire de files d'attente dans une image z/OS différente au sein du sysplex en cas d'échec de z/OS. Les implications réseau du redémarrage d' IBM MQ ARM sur une autre image z/OS sont décrites dans [«Utilisation d'ARM dans un réseau IBM MQ», à la page 436](#).

Pour activer le redémarrage automatique:

- Configurez un fichier de couple ARM.
- Définir les actions de redémarrage automatique que z/OS doit exécuter dans une *règle ARM*.
- Démarrer la règle ARM.

En outre, IBM MQ doit s'enregistrer auprès d'ARM au démarrage (cela se produit automatiquement).

Remarque : Si vous souhaitez redémarrer automatiquement des gestionnaires de files d'attente dans des images z/OS différentes, vous devez définir chaque gestionnaire de files d'attente en tant que sous-système dans chaque image z/OS sur laquelle ce gestionnaire de files d'attente peut être redémarré, avec un nom de sous-système à quatre caractères unique au niveau du sysplex.

Fichiers de couple ARM

Veillez à définir les fichiers de couple requis pour ARM, et à ce qu'ils soient en ligne et actifs avant de démarrer un gestionnaire de files d'attente pour lequel vous souhaitez la prise en charge d'ARM. L'enregistrement ARM automatique de IBM MQ échoue si les fichiers de couple ne sont pas disponibles au démarrage du gestionnaire de files d'attente. Dans cette situation, IBM MQ suppose que l'absence du fichier couplé signifie que vous ne souhaitez pas la prise en charge d'ARM et que l'initialisation se poursuit.

Pour plus d'informations sur les fichiers de couple ARM, voir [z/OS MVS Setting up a Sysplex](#).

z/OS Règles ARM

Les règles du gestionnaire de redémarrage automatique sont des règles définies par l'utilisateur qui contrôlent les fonctions ARM pouvant contrôler les redémarrages d'un gestionnaire de files d'attente.

Les fonctions ARM sont contrôlées par une *règle ARM* définie par l'utilisateur. Chaque image z/OS exécutant une instance de gestionnaire de files d'attente qui doit être redémarrée par ARM doit être connectée à un ensemble de données de couple ARM avec une règle ARM active.

IBM fournit une règle ARM par défaut. Vous pouvez définir de nouvelles règles ou remplacer les valeurs par défaut des règles à l'aide de l'interface IXCMIAPU (*utilitaire de données d'administration*) fournie avec z/OS. [z/OS MVS Setting up a Sysplex](#) décrit cet utilitaire et inclut des détails complets sur la définition d'une règle ARM.

La [Figure 53](#), à la page 435 illustre un exemple de règle ARM. Cet exemple de règle redémarre tout gestionnaire de files d'attente au sein d'un sysplex, en cas d'échec du gestionnaire de files d'attente ou d'un système complet.

```
//IXCMIAPU EXEC PGM=IXCMIAPU,REGION=2M
//SYSPRINT DD SYSOUT=*
//SYSIN DD *
DATA TYPE(ARM)
DEFINE POLICY NAME(ARMPOL1) REPLACE(YES)
RESTART_GROUP(DEFAULT)
ELEMENT(*)
RESTART_ATTEMPTS(0) /* Jobs not to be restarted by ARM */
RESTART_GROUP(GROUP1)
ELEMENT(SYSQMGRMQ*) /* These jobs to be restarted by ARM */
/*
```

Figure 53. Exemple de règle ARM

Pour plus d'informations, voir :

- [Définition d'une règle ARM](#)
- [Activation d'une règle ARM](#)
- [Enregistrement avec ARM](#)

Définition d'une règle ARM

Configurez votre règle ARM comme suit:

- Définissez des RESTART_GROUPS pour chaque instance de gestionnaire de files d'attente qui contient également des sous-systèmes CICS ou IMS qui se connectent à cette instance de gestionnaire de files d'attente. Si vous utilisez une convention de dénomination de sous-système, vous pouvez utiliser le caractère '?' et '*' caractères génériques dans vos noms d'élément pour définir des RESTART_GROUPS avec un minimum d'effort de définition.
- Indiquez TERMTYPE (ELEMTERM) pour vos initiateurs de canal afin d'indiquer qu'ils seront redémarrés uniquement si l'initiateur de canal a échoué et que l'image z/OS n'a pas échoué.
- Spécifiez TERMTYPE (ALLTERM) pour vos gestionnaires de files d'attente afin d'indiquer qu'ils seront redémarrés si le gestionnaire de files d'attente a échoué ou si l'image z/OS a échoué.

- Indiquez RESTART_METHOD (BOTH, PERSISTENT) pour les gestionnaires de files d'attente et les initiateurs de canal. Cela indique à ARM de redémarrer à l'aide du JCL qu'il a sauvegardé (après la résolution des symboles système) lors du dernier démarrage. Il indique à ARM d'effectuer cette opération indépendamment de l'échec de l'élément individuel ou de l'échec de l'image z/OS .
- Acceptez les valeurs par défaut pour toutes les autres options de règle ARM.

Activation d'une règle ARM

Pour démarrer votre règle de gestion du redémarrage automatique, exécutez la commande z/OS suivante:

```
SETXCF START,POLICY,TYPE=ARM,POLNAME= mypol
```

Lorsque la règle est démarrée, tous les systèmes connectés au fichier de couplage ARM utilisent la même règle active.

Utilisez la commande SETXCF STOP pour désactiver les redémarrages automatiques.

Enregistrement avec ARM

IBM MQ s'enregistre automatiquement en tant qu' *élément ARM* lors du démarrage du gestionnaire de files d'attente (sous réserve de la disponibilité d'ARM). Il se désenregistre lors de sa phase d'arrêt, sauf s'il lui est demandé de ne pas le faire.

Au démarrage, le gestionnaire de files d'attente détermine si ARM est disponible. Si tel est le cas, IBM MQ s'enregistre à l'aide du nom SYSMQMGR *ssid*, où *ssid* est le nom du gestionnaire de files d'attente à quatre caractères et SYSMQMGR est le type d'élément.

Les commandes STOP QMGR MODE (QUIESCE) et STOP QMGR MODE (FORCE) désenregistrent le gestionnaire de files d'attente d'ARM (s'il a été enregistré auprès d'ARM au démarrage). Cela empêche ARM de redémarrer ce gestionnaire de files d'attente. La commande STOP QMGR MODE (RESTART) n'a pas pour effet de désenregistrer le gestionnaire de files d'attente du gestionnaire de files d'attente d'ARM. Elle est donc éligible pour un redémarrage automatique immédiat.

Chaque espace adresse d'initiateur de canal détermine si ARM est disponible et, si tel est le cas, s'enregistre avec le nom d'élément SYSMQCH *ssid*, où *ssid* est le nom du gestionnaire de files d'attente et SYSMQCH est le type d'élément.

L'initiateur de canal est toujours désenregistré d'ARM lorsqu'il s'arrête normalement et reste enregistré uniquement s'il se termine de manière anormale. L'initiateur de canal est toujours désenregistré en cas d'échec du gestionnaire de files d'attente.

Utilisation d'ARM dans un réseau IBM MQ

Vous pouvez configurer votre gestionnaire de files d'attente de sorte que les initiateurs de canal et les programmes d'écoute associés soient démarrés automatiquement lors du redémarrage du gestionnaire de files d'attente.

Pour garantir un redémarrage entièrement automatique du gestionnaire de files d'attente sur la même image z/OS pour les protocoles de communication LU 6.2 et TCP/IP:

- Démarrez automatiquement vos programmes d'écoute en ajoutant la commande START LISTENER appropriée au fichier CSQINPX.
- Démarrez automatiquement votre initiateur de canal en ajoutant la commande START CHINIT appropriée au fichier CSQINP2 .

Pour redémarrer un gestionnaire de files d'attente avec TCP/IP ou LU6.2, voir

- [«Redémarrage sur une autre image z/OS avec TCP/IP», à la page 437](#)

- [«Redémarrage sur une autre image z/OS avec LU 6.2», à la page 438](#)

Voir [Tâche 13: Personnalisation des fichiers d'entrée d'initialisation](#) pour plus d'informations sur les fichiers CSQINP2 et CSQINPX.

Redémarrage sur une autre image z/OS avec TCP/IP

Si vous utilisez TCP/IP comme protocole de communication et que vous utilisez des adresses IP virtuelles, vous pouvez les configurer pour une reprise sur d'autres images z/OS, ce qui permet aux canaux qui se connectent à ce gestionnaire de files d'attente de se reconnecter sans aucune modification. Sinon, vous pouvez réallouer une adresse TCP/IP après avoir déplacé un gestionnaire de files d'attente vers une autre image z/OS uniquement si vous utilisez des clusters ou si vous vous connectez à un groupe de partage de files d'attente à l'aide d'un nom de groupe logique DNS (Dynamic Domain Name System) WLM.

- [Lors de l'utilisation de la mise en cluster](#)
- [Lors de la connexion à un groupe de partage de files d'attente](#)

Lors de l'utilisation du groupement

z/OS ARM répond à une défaillance du système en redémarrant le gestionnaire de files d'attente sur une image z/OS différente dans le même sysplex ; ce système possède une adresse TCP/IP différente de l'image z/OS d'origine. La section suivante explique comment utiliser des clusters IBM MQ pour réaffecter l'adresse TCP/IP d'un gestionnaire de files d'attente après son déplacement par redémarrage ARM vers une image z/OS différente.

Lorsqu'un gestionnaire de files d'attente client détecte la défaillance du gestionnaire de files d'attente (en tant que défaillance de canal), il répond en réallouant les messages appropriés sur sa file d'attente de transmission de cluster à un gestionnaire de files d'attente de serveur différent qui héberge une instance différente de la file d'attente de cluster cible. Toutefois, il ne peut pas réallouer les messages liés au serveur d'origine par des contraintes d'affinité ou les messages en attente de validation car le gestionnaire de files d'attente du serveur a échoué lors du traitement de fin de lot. Pour traiter ces messages, procédez comme suit:

1. Allouez un nom de canal récepteur de cluster différent et un port TCP/IP différent à chaque gestionnaire de files d'attente z/OS. Chaque gestionnaire de files d'attente a besoin d'un port différent pour que deux systèmes puissent partager une seule pile TCP/IP sur une image z/OS. L'un d'eux est le gestionnaire de files d'attente qui s'exécute à l'origine sur cette image z/OS, et l'autre est le gestionnaire de files d'attente qu'ARM redémarrera sur cette image z/OS suite à une défaillance du système. Configurez chaque port sur chaque image z/OS afin qu'ARM puisse redémarrer n'importe quel gestionnaire de files d'attente sur n'importe quelle image z/OS.
2. Créez un fichier d'entrée de commande d'initiateur de canal différent (CSQINPX) pour chaque combinaison de gestionnaire de files d'attente et d'image z/OS, à référencer lors du démarrage de l'initiateur de canal.

Chaque fichier CSQINPX doit inclure une commande START LISTENER PORT (port) spécifique à ce gestionnaire de files d'attente et une commande ALTER CHANNEL pour un canal récepteur de cluster spécifique à cette combinaison de gestionnaires de files d'attente et d'images z/OS. La commande ALTER CHANNEL doit définir le nom de connexion sur le nom TCP/IP de l'image z/OS sur laquelle elle est redémarrée. Il doit inclure le numéro de port spécifique au gestionnaire de files d'attente redémarré dans le nom de la connexion.

Le JCL de démarrage de chaque gestionnaire de files d'attente peut avoir un nom de fichier fixe pour ce fichier CSQINPX, et chaque image z/OS doit avoir une version différente de chaque fichier CSQINPX sur un volume DASD non partagé.

Si un redémarrage ARM se produit, IBM MQ annonce la définition de canal modifiée dans le référentiel de cluster, qui à son tour la publie sur tous les gestionnaires de files d'attente client qui ont exprimé un intérêt pour le gestionnaire de files d'attente du serveur.

Le gestionnaire de files d'attente client traite la défaillance du gestionnaire de files d'attente du serveur comme une défaillance de canal et tente de redémarrer le canal défaillant. Lorsque le

gestionnaire de files d'attente client apprend le nouveau nom de connexion serveur, le redémarrage du canal reconnecte le gestionnaire de files d'attente client au gestionnaire de files d'attente serveur redémarré. Le gestionnaire de files d'attente client peut ensuite resynchroniser ses messages, résoudre les messages en attente de validation dans la file d'attente de transmission du gestionnaire de files d'attente client et le traitement normal peut se poursuivre.

Lors de la connexion à un groupe de partage de files d'attente

Lors de la connexion à un groupe de partage de files d'attente via un nom de groupe logique DNS (Domain Name System) dynamique TCP/IP, le nom de connexion dans votre définition de canal indique le nom de groupe logique de votre groupe de partage de files d'attente, et non le nom d'hôte ou l'adresse IP d'une machine physique. Lorsque ce canal démarre, il se connecte au serveur DNS dynamique, puis est connecté à l'un des gestionnaires de files d'attente du groupe de partage de files d'attente. Ce processus est expliqué dans la rubrique [Configuration de la communication pour IBM MQ for z/OS à l'aide de groupes de partage de files d'attente](#).

Dans le cas peu probable d'un échec d'image, l'une des situations suivantes se produit:

- Les gestionnaires de files d'attente sur le désenregistrement de l'image défaillante du serveur de noms de domaine dynamique en cours d'exécution sur votre sysplex. Le canal répond à l'échec de la connexion en passant à l'état REESSAYEZ, puis se connecte au serveur de noms de domaine dynamique qui s'exécute sur le sysplex. Le serveur de noms de domaine dynamique alloue la demande entrante à l'un des membres restants du groupe de partage de files d'attente qui est toujours en cours d'exécution sur les images restantes.
- Si aucun autre gestionnaire de files d'attente du groupe de partage de files d'attente n'est actif et qu'ARM redémarre le gestionnaire de files d'attente et l'initiateur de canal sur une image différente, le programme d'écoute de groupe s'enregistre avec le serveur de noms de domaine dynamique à partir de cette nouvelle image. Cela signifie que le nom de groupe logique (à partir de la zone de nom de connexion du canal) se connecte au serveur de noms de domaine dynamique, puis est connecté au même gestionnaire de files d'attente, s'exécutant désormais sur une image différente. Aucune modification n'a été requise pour la définition de canal.

Pour que ce type de récupération se produise, les points suivants doivent être notés:

- Sous z/OS, le serveur de noms de domaine dynamique s'exécute sur l'une des images z/OS du sysplex. Si cette image échoue, le serveur de noms de domaine dynamique doit être configuré pour qu'un serveur de noms secondaire soit actif dans le sysplex, en tant qu'alternative au serveur de noms principal. Des informations sur les serveurs DNS dynamiques primaires et secondaires sont disponibles dans le manuel [OS/390 SecureWay CS IP Configuration](#).
- Le programme d'écoute de groupe TCP/IP a peut-être été démarré sur une adresse IP particulière qui n'est peut-être pas disponible sur cette image z/OS. Si tel est le cas, il se peut que le programme d'écoute doive être démarré sur une adresse IP différente sur la nouvelle image. Si vous utilisez des adresses IP virtuelles, vous pouvez les configurer pour une reprise sur d'autres images z/OS de sorte qu'aucune modification de la commande START LISTENER ne soit requise.

Redémarrage sur une autre image z/OS avec LU 6.2

Si vous utilisez uniquement les protocoles de communication LU 6.2, procédez comme suit pour activer la reconnexion réseau après le redémarrage automatique d'un gestionnaire de files d'attente sur une image z/OS différente dans le sysplex:

- Définissez chaque gestionnaire de files d'attente dans le sysplex avec un nom de sous-système unique.
- Définissez chaque initiateur de canal dans le sysplex avec un nom d'unité logique unique. Cette valeur est spécifiée à la fois dans les attributs du gestionnaire de files d'attente et dans la commande START LISTENER.

Remarque : Le nom d'unité logique (LUNAME) désigne une entrée dans la table secondaire APPC, qui à son tour mappe cette entrée vers le nom d'unité logique réel.

- Configurez une table secondaire APPC partagée, qui est référencée par chaque image z/OS dans le sysplex. Il doit contenir une entrée pour le nom d'unité logique de chaque initiateur de canal. Pour plus d'informations, voir *z/OS MVS Planning: APPC/MVS Management* .
- Configurez un membre APPCPM xx de SYS1.PARMLIB pour chaque initiateur de canal dans le sysplex devant contenir une unité de travail logique (LUADD) pour activer l'entrée de table côté APPC pour cet initiateur de canal. Ces membres doivent être partagés par chaque image z/OS . Le système SYS1.PARMLIB est activé par une commande z/OS SET APPC= xx, qui est émise automatiquement lors du redémarrage ARM du gestionnaire de files d'attente (et de son initiateur de canal) sur une image z/OS différente, comme décrit dans le texte suivant.
- Utilisez l'attribut de gestionnaire de files d'attente LU62ARM pour spécifier le suffixe xx de ce SYS1.PARMLIB pour chaque initiateur de canal. L'initiateur de canal émet alors la commande z/OS SET APPC= xx requise pour activer son nom d'unité logique (LUNAME).

Définissez votre règle ARM de sorte qu'elle redémarre l'initiateur de canal uniquement en cas d'échec de son image z/OS ; l'ID utilisateur associé à l'espace adresse XCFAS doit être autorisé à émettre la commande IBM MQ START CHINIT. Ne redémarrez pas automatiquement l'initiateur de canal si son image z/OS échoue également. Utilisez à la place les commandes des fichiers CSQINP2 et CSQINPX pour démarrer l'initiateur de canal et les programmes d'écoute.

Récupération manuelle des unités de travail

Vous pouvez récupérer manuellement des unités d'oeuvre CICS, IMS, RRS ou d'autres gestionnaires de files d'attente dans un groupe de partage de files d'attente. Vous pouvez utiliser les commandes du gestionnaire de files d'attente pour afficher le statut des unités d'oeuvre associées à chaque connexion au gestionnaire de files d'attente.

Cette rubrique contient des informations sur les sujets suivants:

- [«Affichage des connexions et des unités d'exécution», à la page 439](#)
- [«Récupération manuelle des unités de récupération CICS», à la page 440](#)
- [«Récupération manuelle des unités de récupération IMS», à la page 443](#)
- [«Récupération manuelle des unités RRS de récupération», à la page 445](#)
- [«Récupération d'unités de récupération sur un autre gestionnaire de files d'attente dans le groupe de partage de files d'attente», à la page 446](#)

Affichage des connexions et des unités d'exécution

Vous pouvez utiliser la commande `DISPLAY CONN` pour obtenir des informations sur les connexions aux gestionnaires de files d'attente et les unités d'oeuvre associées. Vous pouvez afficher les unités d'oeuvre actives pour voir ce qui se passe actuellement, ou pour voir ce qui doit être arrêté pour permettre au gestionnaire de files d'attente de s'arrêter, et vous pouvez afficher les unités d'oeuvre non résolues pour faciliter la reprise.

Unités de travail actives

Pour afficher uniquement les unités de travail actives, utilisez

```
DISPLAY CONN(*) WHERE(UOWSTATE EQ ACTIVE)
```

Unités de travail non résolues

Une unité d'oeuvre non résolue, également appelée "unité d'exécution en attente de validation", est une unité qui se trouve dans le deuxième passage de l'opération de validation en deux phases. Les

ressources sont conservées dans IBM MQ en son nom. Pour afficher les unités de travail non résolues, utilisez

```
DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Une intervention externe est nécessaire pour résoudre le statut des unités d'oeuvre non résolues. Cela peut impliquer uniquement le démarrage du coordinateur de reprise (CICS, IMSou RRS) ou en impliquer davantage, comme décrit dans les sections suivantes.

Récupération manuelle des unités de récupération CICS

Utilisez cette rubrique pour comprendre ce qui se passe lorsque l'adaptateur CICS redémarre, puis explique comment traiter les unités de récupération non résolues qui se produisent.

Que se passe-t-il lorsque l'adaptateur CICS redémarre?

Chaque fois qu'une connexion est interrompue, l'adaptateur doit passer par une *phase de redémarrage* lors du *processus de reconnexion*. La phase de redémarrage resynchronise les ressources. La resynchronisation entre CICS et IBM MQ permet d'identifier et de résoudre les unités d'oeuvre en attente de validation.

La resynchronisation peut être provoquée par:

- Demande explicite du composant de mise en file d'attente répartie
- Une demande implicite lorsqu'une connexion est établie à IBM MQ

Si la resynchronisation est provoquée par la connexion à IBM MQ, la séquence d'événements est la suivante:

1. Le processus de connexion extrait une liste d'ID d'unités de travail (UOW) en attente de validation de IBM MQ.
2. Les ID unité de travail sont affichés sur la console dans les messages CSQC313I .
3. Les ID unité de travail sont transmis à CICS.
4. CICS lance une tâche de resynchronisation (CRSY) pour chaque ID unité de travail en attente de validation.
5. Le résultat de la tâche pour chaque unité de travail en attente de validation s'affiche sur la console.

Vous devez vérifier les messages affichés lors du processus de connexion:

CSQC313I

Indique qu'une unité de travail est en attente de validation.

CSQC400I

Identifie l'unité de travail et est suivi de l'un des messages suivants:

- CSQC402I ou CSQC403I indique que l'unité de travail a été résolue avec succès (validée ou annulée).
- CSQC404E, CSQC405E, CSQC406E ou CSQC407E indique que l'unité de travail n'a pas été résolue.

CSQC409I

Indique que toutes les unités de travail ont été résolues avec succès.

CSQC408I

Indique que toutes les unités de travail n'ont pas été résolues correctement.

CSQC314I

Avertit que les ID unité de travail mis en évidence avec un astérisque (*) ne sont pas résolus automatiquement. Ces unités de travail doivent être résolues explicitement par le composant de mise en file d'attente répartie lors de son redémarrage.

La [Figure 54](#), à la [page 441](#) illustre un exemple d'ensemble de messages de redémarrage affichés sur la console z/OS .


```

CSQ9022I +CSQ1 CSQYASCP ' START QMGR' NORMAL COMPLETION
+CSQC323I VICIC1 CSQCQCON CONNECT received from TERMID=PB62 TRANID=CKCN
+CSQC303I VICIC1 CSQCCON CSQCSERV loaded. Entry point is 850E8918
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F0E2178D25 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F055B2AC25 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6EFFD60D425 is in doubt
+CSQC313I VICIC1 CSQCCON UOWID=VICIC1.A6E5A6F07AB56D22 is in doubt
+CSQC307I VICIC1 CSQCCON Successful connection to subsystem VC2
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BAD18) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BAA10) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008BA708) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CAE88) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CAB80) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA878) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA570) connect
successful
+CSQC472I VICIC1 CSQCSERV Server subtask (TCB address=008CA268) connect
successful
+CSQC403I VICIC1 CSQCTRU Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRU UOWID=VICIC1.A6E5A6F0E2178D25
+CSQC403I VICIC1 CSQCTRU Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRU UOWID=VICIC1.A6E5A6F055B2AC25
+CSQC403I VICIC1 CSQCTRU Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRU UOWID=VICIC1.A6E5A6F07AB56D22
+CSQC403I VICIC1 CSQCTRU Resolved BACKOUT for
+CSQC400I VICIC1 CSQCTRU UOWID=VICIC1.A6E5A6EFFD60D425
+CSQC409I VICIC1 CSQCTRU Resynchronization completed successfully

```

Figure 54. Exemples de messages de redémarrage

Le nombre total de messages CSQC313I doit être égal au nombre total de messages CSQC402I plus CSQC403I . Si les totaux ne sont pas égaux, il existe des unités de travail que le processus de connexion ne peut pas résoudre. Les unités de travail qui ne peuvent pas être résolues sont dues à des problèmes liés à CICS (par exemple, un démarrage à froid) ou à IBM MQ, ou à la distribution de la mise en file d'attente. Une fois ces problèmes résolus, vous pouvez lancer une autre resynchronisation en vous déconnectant, puis en vous reconnectant.

Vous pouvez également résoudre chaque unité de travail en attente à l'aide de la commande RESOLVE INDOUBT et de l'ID unité de travail indiqué dans le message CSQC400I. Vous devez ensuite initier une déconnexion et une connexion pour nettoyer l' *unité de descripteurs de reprise* dans CICS. Vous devez connaître le résultat correct de l'unité de travail pour résoudre les unités de travail manuellement.

Tous les messages associés à des unités de travail non résolues sont verrouillés par IBM MQ et aucune tâche par lots, TSO ou CICS ne peut y accéder.

Si CICS échoue et qu'un redémarrage d'urgence est nécessaire, *ne pas* modifier l'identificateur d'application GENERIC du système CICS . Si vous le faites, puis vous reconnectez à IBM MQ, l'intégrité des données avec IBM MQ ne peut pas être garantie. En effet, IBM MQ traite la nouvelle instance de CICS comme un CICS différent (car l'identificateur d'application est différent). La résolution en attente de validation est alors basée sur le mauvais journal CICS .

Comment résoudre les unités de récupération CICS manuellement

Si l'adaptateur s'arrête de manière anormale, CICS et IBM MQ génèrent des listes en attente de validation de manière dynamique ou lors du redémarrage, en fonction du sous-système à l'origine de la fin anormale.

Remarque : Si vous utilisez l'exemple de programme DFH\$INDB pour afficher les unités de travail, vous pouvez constater qu'il n'affiche pas toujours correctement les unités de travail IBM MQ .

Lorsque CICS se connecte à IBM MQ, il se peut qu'une ou plusieurs unités de récupération n'aient pas été résolues.

L'un des messages suivants est envoyé à la console:

- CSQC404E
- CSQC405E
- CSQC406E
- CSQC407E
- CSQC408I

Pour plus de détails sur la signification de ces messages, voir les messages [Messages d'adaptateur et de pont CICS](#).

CICS conserve les détails des unités de récupération qui n'ont pas été résolues lors du démarrage de la connexion. Une entrée est purgée lorsqu'elle n'apparaît plus dans la liste présentée par IBM MQ.

Les unités de récupération que CICS ne peut pas résoudre doivent être résolues manuellement à l'aide des commandes IBM MQ. Cette procédure manuelle est rarement utilisée dans une installation, car elle n'est requise que lorsque des erreurs opérationnelles ou des problèmes logiciels ont empêché la résolution automatique. *Toute incohérence détectée lors de la résolution en attente de validation doit être examinée.*

Pour résoudre les unités de récupération:

1. Obtenez la liste des unités de récupération à partir de IBM MQ à l'aide de la commande suivante:

```
+CSQ1 DISPLAY CONN( * ) WHERE(UOWSTATE EQ UNRESOLVED)
```

Le message suivant s'affiche :

```
CSQM201I +CSQ1 CSQMDRTC DISPLAY CONN DETAILS
CONN(BC85772CBE3E0001)
EXTCONN(C3E2D8C3C7D9F0F940404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2005-02-04)
UOWLOGTI(10.17.44)
UOWSTDA(2005-02-04)
UOWSTTI(10.17.44)
UOWSTATE(UNRESOLVED)
NID(IYRCSQ1 .BC8571519B60222D)
EXTURID(BC8571519B60222D)
QMURID(0000002BDA50)
URTYPE(CICS)
USERID(MQTEST)
APPLTAG(IYRCSQ1)
ASID(0000)
APPLTYPE(CICS)
TRANSID(GP02)
TASKNO(0000096)
END CONN DETAILS
```

Pour les connexions CICS, le NID se compose de l'ID d'application CICS et d'un numéro unique fourni par CICS lors de l'écriture des entrées de journal de point de synchronisation. Ce numéro unique est stocké dans les enregistrements écrits à la fois dans le journal système CICS et dans le journal IBM MQ au moment du traitement du point de synchronisation. Cette valeur est appelée *jeton de récupération* dans CICS.

2. Recherchez dans le journal CICS les entrées associées à une unité de récupération particulière.

Recherchez un enregistrement PREPARE pour l'installation liée à la tâche où la zone JCSRMTKN (jeton de récupération) est égale à la valeur obtenue à partir de l'ID réseau. L'ID réseau est fourni par IBM MQ dans la sortie de la commande DISPLAY CONN.

L'enregistrement PREPARE du journal CICS des unités de récupération indique le numéro de tâche CICS . Toutes les autres entrées du journal de cette tâche CICS peuvent être localisées à l'aide de ce nombre.

Vous pouvez utiliser l'utilitaire d'impression de journal CICS DFHJUP lors de l'analyse du journal. Pour plus d'informations sur l'utilisation de ce programme, voir le manuel *CICS Operations and Utilities Guide*.

3. Analysez le journal IBM MQ à la recherche d'enregistrements avec le NID associé à une unité de récupération particulière. Utilisez ensuite l'URID de cet enregistrement pour obtenir le reste des enregistrements de journal pour cette unité de récupération.

Lors de l'analyse du journal IBM MQ , notez que le IBM MQ message de démarrage CSQJ001I fournit l'adresse relative en octets de début pour cette session.

Le programme d'impression des enregistrements de journal (CSQ1LOGP) peut être utilisé à cette fin.

4. Si nécessaire, effectuez une résolution en attente de validation dans IBM MQ.

IBM MQ peut être invité à effectuer l'action de reprise pour une unité de récupération à l'aide d'une commande IBM MQ [RESOLVE INDOUBT](#) .

Pour récupérer toutes les unités d'exécution associées à un *nom de connexion* spécifique, utilisez l'option NID (*).

La commande génère l'un des messages suivants indiquant si l'unité d'exécution est validée ou annulée:

```
CSQV414I +CSQ1 THREAD network-id COMMIT SCHEDULED
CSQV415I +CSQ1 THREAD network-id ABORT SCHEDULED
```

Lors de l'exécution de la résolution en attente de validation, CICS et l'adaptateur ne prennent pas en compte les commandes de IBM MQ permettant de valider ou d'éliminer les unités de récupération, car seules les ressources IBM MQ sont affectées. Toutefois, CICS conserve des détails sur les unités d'exécution en attente de validation qui n'ont pas pu être résolues par IBM MQ. Ces informations sont purgées lorsque la liste présentée est vide ou lorsque la liste ne contient pas d'unité de récupération dont CICS dispose de détails.

Récupération manuelle des unités de récupération IMS

Utilisez cette rubrique pour comprendre ce qui se passe lorsque l'adaptateur IMS redémarre, puis explique comment traiter les unités de récupération non résolues qui se produisent.

Que se passe-t-il lorsque l'adaptateur IMS redémarre?

Chaque fois que la connexion à IBM MQ est redémarrée, à la suite d'un redémarrage du gestionnaire de files d'attente ou d'une commande IMS /START SUBSYS, IMS lance le processus de resynchronisation suivant:

1. Le IMS présente la liste des ID unité de travail (UOW) qui, selon lui, sont en attente de validation pour l'adaptateur IBM MQ IMS , un par un, avec le paramètre de résolution Validation ou Annulation.
2. L'adaptateur IMS transmet la demande de résolution à IBM MQ et renvoie le résultat à IMS.
3. Après avoir traité toutes les demandes de résolution IMS , l'adaptateur IMS obtient de IBM MQ la liste de toutes les unités de travail que IBM MQ détient toujours en attente de validation et qui ont été lancées par le système IMS . Ils sont signalés au terminal maître IMS dans le message CSQQ008I.

Remarque : Lorsqu'une unité de travail est en attente de validation, tout message IBM MQ associé est verrouillé par IBM MQ et n'est disponible pour aucune application.

Comment résoudre les unités de récupération IMS manuellement

Lorsque IMS se connecte à IBM MQ, IBM MQ peut avoir une ou plusieurs unités de récupération en attente de validation qui n'ont pas été résolues.

Si IBM MQ possède des unités de récupération en attente de validation qui n'ont pas été résolues par IMS, le message suivant est émis sur le terminal maître IMS :

```
CSQQ008I nn units of recovery are still in doubt in queue manager qmgr-name
```

Si ce message est émis, cela signifie que IMS a été démarré à froid ou qu'il a été démarré avec une bande de journal incomplète. Ce message peut également être émis si IBM MQ ou IMS s'arrête de manière anormale en raison d'une erreur logicielle ou d'une autre défaillance du sous-système.

Après réception du message CSQQ008I :

- La connexion reste active.
- Les applications IMS peuvent toujours accéder aux ressources IBM MQ.
- Certaines ressources IBM MQ restent verrouillées.

Si l'unité d'exécution en attente de validation n'est pas résolue, les files d'attente de messages IMS peuvent commencer à s'accumuler. Si les files d'attente IMS sont saturées, IMS s'arrête. Vous devez être conscient de cette difficulté potentielle et surveiller IMS jusqu'à ce que les unités de récupération en attente de validation soient complètement résolues.

Procédure de récupération

Utilisez la procédure suivante pour récupérer les unités d'oeuvre IMS :

1. Forcez la fermeture du journal IMS à l'aide du fichier journal en ligne /SWI, puis archivez le journal IMS. Utilisez l'utilitaire DFSERA10 pour imprimer les enregistrements de la bande de journal IMS précédente. Les enregistrements de journal de type X'3730' indiquent une demande de validation phase-2 et les enregistrements de journal de type X'38' indiquent une demande d'abandon. Enregistrez l'action demandée pour la dernière transaction dans chaque région dépendante.
2. Exécutez le travail par lots DL/I pour rétablir chaque bloc de spécification de programme impliqué qui n'a pas atteint un point de validation. Le processus peut prendre un certain temps car les transactions sont toujours en cours de traitement. Il peut également verrouiller un certain nombre d'enregistrements, ce qui peut affecter le reste du traitement et le reste des files d'attente de messages.
3. Générez une liste des unités de récupération en attente de validation à partir de IBM MQ à l'aide de la commande suivante:

```
+CSQ1 DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

Le message suivant s'affiche :

```
CSQM201I +CSQ1 CSQMDRTC DISPLAY CONN DETAILS
CONN(BC45A794C4290001)
EXTCONN(C3E2D8C3E2C5C3F240404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2005-02-15)
UOWLOGTI(16.39.43)
```

```

UOWSTDA (2005-02-15)
UOWSTTI (16.39.43)
UOWSTATE (UNRESOLVED)
NID (IM8F .BC45A794D3810344)
EXTURID (
0000052900000000
)
QMURID (00000354B76E)
URTYPE (IMS)
USERID (STCPI)
APPLTAG (IM8F)
ASID (0000)
APPLTYPE (IMS)
PSTID (0004)
PSBNAME (GP01MPP)

```

Pour IMS, le NID se compose du nom de connexion IMS et d'un numéro unique fourni par IMS. La valeur est appelée *jeton de récupération* dans IMS . Pour plus d'informations, voir *IMS -Guide de personnalisation*.

4. Comparez les NID (IMSID plus OASN en hexadécimal) affichés dans les messages DISPLAY THREAD avec les OASN (4 octets décimaux) affichés dans la sortie DFSERA10 . Choisissez de valider ou d'abandonner.
5. Effectuez la résolution en attente de validation dans IBM MQ à l'aide de la commande RESOLVE INDOUBT , comme suit:

```

RESOLVE INDOUBT( connection-name )
ACTION(COMMIT|BACKOUT)
NID( network-id )

```

Pour récupérer toutes les unités d'exécution associées à *connection-name*, utilisez l'option NID (*). La commande génère l'un des messages suivants pour indiquer si l'unité d'exécution est validée ou annulée:

```

CSQV414I  THREAD network-id COMMIT SCHEDULED
CSQV415I  THREAD network-id BACKOUT SCHEDULED

```

Lors de l'exécution de la résolution en attente de validation, IMS et l'adaptateur n'ont pas connaissance des commandes permettant à IBM MQ de valider ou d'éliminer les unités de récupération en attente de validation car seules les ressources IBM MQ sont affectées.

Récupération manuelle des unités RRS de récupération

Cette rubrique explique comment déterminer s'il existe des unités de récupération RRS en attente de validation et comment les résoudre manuellement.

Lorsque RRS se connecte à IBM MQ, IBM MQ peut avoir une ou plusieurs unités de récupération en attente de validation qui n'ont pas été résolues. Si IBM MQ possède des unités de récupération en attente de validation que RRS n'a pas résolues, l'un des messages suivants est émis sur la console z/OS :

- CSQ3011I
- CSQ3013I
- CSQ3014I
- CSQ3016I

IBM MQ et RRS fournissent des outils permettant d'afficher des informations sur les unités de récupération en attente de validation et des techniques permettant de les résoudre manuellement.

Dans IBM MQ, utilisez la commande DISPLAY CONN pour afficher des informations sur les unités d'exécution IBM MQ en attente de validation. La sortie de la commande inclut des ID d'unité de récupération RRS pour les unités d'exécution IBM MQ ayant RRS comme coordinateur. Il peut être utilisé pour déterminer le résultat de l'unité de récupération.

Utilisez la commande RESOLVE INDOUBT pour résoudre manuellement l'unité d'exécution en attente de validation IBM MQ . Cette commande peut être utilisée pour valider ou restaurer l'unité de récupération après avoir déterminé la décision correcte.

z/OS **Récupération d'unités de récupération sur un autre gestionnaire de files d'attente dans le groupe de partage de files d'attente**

Cette rubrique permet d'identifier et de récupérer manuellement des unités de récupération sur d'autres gestionnaires de files d'attente dans un groupe de partage de files d'attente.

Si un gestionnaire de files d'attente membre d'un groupe de partage de files d'attente échoue et ne peut pas être redémarré, les autres gestionnaires de files d'attente du groupe peuvent effectuer une reprise homologue et en prendre le relais. Toutefois, le gestionnaire de files d'attente peut avoir des unités de récupération en attente de validation qui ne peuvent pas être résolues par la reprise homologue car la disposition finale de cette unité de récupération est connue uniquement du gestionnaire de files d'attente défaillant. Ces unités de récupération sont résolues lorsque le gestionnaire de files d'attente est finalement redémarré, mais jusqu'à ce moment-là, elles restent en attente de validation.

Cela signifie que certaines ressources (par exemple, les messages) peuvent être verrouillées, ce qui les rend indisponibles pour d'autres gestionnaires de files d'attente du groupe. Dans cette situation, vous pouvez utiliser la commande DISPLAY THREAD pour afficher ces unités d'oeuvre sur le gestionnaire de files d'attente inactif. Si vous souhaitez résoudre ces unités de récupération manuellement afin de rendre les messages disponibles pour les autres gestionnaires de files d'attente du groupe, vous pouvez utiliser la commande RESOLVE INDOUBT.

Lorsque vous émettez la commande DISPLAY THREAD pour afficher les unités de récupération en attente de validation, vous pouvez utiliser le mot clé QMNAME pour indiquer le nom du gestionnaire de files d'attente inactif. Par exemple, si vous exécutez la commande suivante:

```
+CSQ1 DISPLAY THREAD(*) TYPE(INDOUBT) QMNAME(QM01)
```

Vous recevez les messages suivants:

```
CSQV436I +CSQ1 INDOUBT THREADS FOR QM01 -
NAME  THREAD-XREF  URID  NID
USER1  00000000000000000000000000000000 CSQ:0001.0
USER2  00000000000000000000000000000000 CSQ:0002.0
DISPLAY THREAD REPORT COMPLETE
```

Si le gestionnaire de files d'attente spécifié est actif, IBM MQ ne renvoie pas d'informations sur les unités d'exécution en attente de validation, mais émet le message suivant:

```
CSQV435I CANNOT USE QMNAME KEYWORD, QM01 IS ACTIVE
```

Utilisez la commande IBM MQ RESOLVE INDOUBT pour résoudre manuellement les unités d'exécution en attente de validation. Utilisez le mot clé QMNAME pour spécifier le nom du gestionnaire de files d'attente inactif dans la commande.

Cette commande peut être utilisée pour valider ou restaurer l'unité de récupération. La commande résout la partie partagée de l'unité de récupération uniquement ; tous les messages locaux ne sont pas affectés et restent verrouillés jusqu'à ce que le gestionnaire de files d'attente redémarre ou se reconnecte à CICS, IMSou RRS par lots.

z/OS IBM MQ et IMS

IBM MQ fournit deux composants pour l'interface avec IMS, la carte IBM MQ - IMS et le pont IBM MQ - IMS . Ces composants sont communément appelés adaptateur IMS et pont IMS .

z/OS Utilisation de l'adaptateur IMS

Cette rubrique explique comment utiliser l'adaptateur IMS , qui connecte IBM MQ aux systèmes IMS .

Remarque : L'adaptateur IMS n'intègre aucun panneau d'opérations et de contrôle.

Cette rubrique contient les sections suivantes :

- [«Contrôle des connexions IMS», à la page 447](#)
- [«Connexion à partir de la région de contrôle IMS», à la page 447](#)
- [«Affichage des unités de récupération en attente de validation», à la page 449](#)
- [«Contrôle des connexions de région dépendante IMS», à la page 451](#)
- [«déconnexion deIMS», à la page 454](#)
- [«Contrôle du moniteur de déclenchement IMS», à la page 454](#)

z/OS Contrôle des connexions IMS

Utilisez cette rubrique pour comprendre les commandes de l'opérateur IMS qui contrôlent et surveillent la connexion à IBM MQ.

IMS fournit les commandes d'opérateur suivantes pour contrôler et surveiller la connexion à IBM MQ:

/MODIFIER SOUS-SYSTEME

Supprime une unité de récupération en attente de validation de IMS.

/DISPLAY SOUS-SYSTEME OASN

Affiche les éléments de reprise en attente.

/DISPLAY SOUS-SYSTEME

Affiche le statut de connexion et l'activité des unités d'exécution.

/DEBUT SOUS-SYSTEME

Connecte la région de contrôle IMS à un gestionnaire de files d'attente.

/ARRET SOUS-SYSTEME

Déconnecte IMS d'un gestionnaire de files d'attente.

/TRACE

Contrôle la trace IMS .

Pour plus d'informations sur ces commandes, voir le manuel *IMS/ESA Operator's Reference* pour le niveau de IMS que vous utilisez.

Les réponses de commande IMS sont envoyées au terminal à partir duquel la commande a été émise. L'autorisation d'émettre des commandes IMS est basée sur la sécurité IMS .

z/OS Connexion à partir de la région de contrôle IMS

Utilisez cette rubrique pour comprendre les mécanismes disponibles pour la connexion de IMS à IBM MQ.

IMS établit une connexion à partir de sa région de contrôle à chaque gestionnaire de files d'attente qui utilise IMS. IMS doit être activé pour établir la connexion de l'une des manières suivantes:

- Automatiquement lors de l'une des opérations suivantes:
 - Initialisation d'un démarrage à froid.
 - Démarrage à chaud de IMS, si la connexion IBM MQ était active lors de l'arrêt de IMS .
- En réponse à la commande IMS :

```
/START SUBSYS sysid
```

où *sysid* est le nom du gestionnaire de files d'attente.

La commande peut être émise que le gestionnaire de files d'attente soit actif ou non.

La connexion n'est pas établie tant que le premier appel d'API MQ au gestionnaire de files d'attente n'a pas été effectué. Jusqu'à ce moment, la commande IMS /DIS SUBSYS affiche le statut 'NOT CONN'.

L'ordre dans lequel vous démarrez IMS et le gestionnaire de files d'attente n'est pas significatif.

IMS ne peut pas réactiver automatiquement la connexion au gestionnaire de files d'attente si ce dernier est arrêté à l'aide de la commande STOP QMGR, de la commande IMS /STOP SUBSYS ou d'un arrêt anormal. Par conséquent, vous devez établir la connexion à l'aide de la commande IMS /START SUBSYS.

Initialisation de l'adaptateur et connexion au gestionnaire de files d'attente

L'adaptateur est un ensemble de modules chargés dans les régions de contrôle et dépendantes IMS, à l'aide de la fonction de connexion de sous-système externe IMS.

Cette procédure initialise l'adaptateur et se connecte au gestionnaire de files d'attente:

1. Lisez le membre de sous-système (SSM) à partir de IMS.PROCLIB. Le SSM choisi est un paramètre IMS EXEC. Il existe une entrée dans le membre pour chaque gestionnaire de files d'attente auquel IMS peut se connecter. Chaque entrée contient des informations de contrôle sur un adaptateur IBM MQ.
2. Chargez l'adaptateur IMS.

Remarque : IMS charge une copie des modules d'adaptateur pour chaque instance IBM MQ définie dans le membre SSM.

3. Connectez la tâche de sous-système externe pour IBM MQ.
4. Exécutez l'adaptateur avec le paramètre CTL EXEC (IDIMS) comme nom de connexion.

Le processus est le même que la connexion fasse partie de l'initialisation ou qu'elle soit le résultat de la commande IMS /START SUBSYS.

Si le gestionnaire de files d'attente est actif lorsque IMS tente d'établir la connexion, les messages suivants sont envoyés:

- dans la console z/OS :

```
DFS3613I ESS TCB INITIALIZATION COMPLETE
```

- au terminal maître IMS :

```
CSQQ000I IMS/TM imsid connected to queue manager ssnm
```

Lorsque IMS tente d'établir la connexion et que *le gestionnaire de files d'attente n'est pas actif*, les messages suivants sont envoyés au terminal maître IMS chaque fois qu'une application effectue un appel MQI:


```
CSQQ001I IMS/TM imsid not connected to queue manager ssnm.  
Notify message accepted  
DFS3607I MQM1 SUBSYSTEM ID EXIT FAILURE, FC = 0286, RC = 08,  
JOBNAME = IMSEMPR1
```

Si vous obtenez des messages DFS3607I lorsque vous démarrez la connexion à IMS ou au démarrage du système, cela indique que le gestionnaire de files d'attente n'est pas disponible. Pour empêcher la génération d'un grand nombre de messages, vous devez effectuer l'une des opérations suivantes:

1. Démarrez le gestionnaire de files d'attente approprié.
2. Exécutez la commande IMS :

```
/STOP SUBSYS
```

afin que IMS ne s'attende pas à se connecter au gestionnaire de files d'attente.

Si vous n'effectuez aucune de ces opérations, un message DFS3607I et le message CSQQ001I associé sont émis chaque fois qu'un travail est planifié dans la région et chaque fois qu'une demande de connexion au gestionnaire de files d'attente est effectuée par une application.

Connexion d'unité d'exécution

Dans une région MPP ou IFP, IMS établit une connexion d'unité d'exécution lorsque le premier programme d'application est planifié dans cette région, même si ce programme d'application n'effectue pas d'appel IBM MQ . Dans une région BMP, la connexion de l'unité d'exécution est établie lorsque l'application effectue son premier appel IBM MQ (MQCONN ou MQCONNX). Cette unité d'exécution est conservée pendant la durée de la région ou jusqu'à l'arrêt de la connexion.

Pour les régions gérées par message et les régions non gérées par message, l'identificateur de référence croisée de l'unité d'exécution de récupération, *Thread-xref*, associé à l'unité d'exécution est:

```
PSTid + PSBname
```

où :

PSTid

Identificateur de région de la table de spécification de partition

PSBname

Nom de bloc de spécifications de programme

Vous pouvez utiliser les ID de connexion en tant qu'identificateurs uniques dans les commandes IBM MQ , auquel cas IBM MQ les insère automatiquement dans tout message d'opérateur qu'il génère.

Affichage des unités de récupération en attente de validation

Vous pouvez afficher les unités de récupération en attente de validation et tenter de les récupérer.

Les étapes opérationnelles utilisées pour répertorier et récupérer les unités de récupération en attente de validation dans cette rubrique ne concernent que les cas relativement simples. Si le gestionnaire de files d'attente s'arrête de manière anormale alors qu'il est connecté à IMS, IMS peut effectuer une validation ou une annulation sans que IBM MQ en ait connaissance. Lorsque le gestionnaire de files d'attente redémarre, le travail est qualifié *en attente de validation*. Une décision doit être prise sur le statut du travail.

Pour afficher la liste des unités de reprise en attente de validation, exécutez la commande suivante:

```
+CSQ1 DISPLAY CONN(*) WHERE(UOWSTATE EQ UNRESOLVED)
```

IBM MQ répond avec un message du type suivant:

```
CSQM201I +CSQ1 CSQMDRTC DIS CONN DETAILS
CONN(BC0F6125F5A30001)
EXTCONN(C3E2D8C3C3E2D8F140404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2004-11-02)
UOWLOGTI(12.27.58)
UOWSTDA(2004-11-02)
UOWSTTI(12.27.58)
UOWSTATE(UNRESOLVED)
NID(CSQ1CHIN.BC0F5F1C86FC0766)
EXTURID(00000000000001F000000007472616E5F6964547565204E6F762020...)
QMURID(0000000026232)
URTYPE(XA)
USERID( )
APPLTAG(CSQ1CHIN)
ASID(0000)
APPLTYPE(CHINIT)
CHANNEL( )
CONNNAME( )
END CONN DETAILS
```

Pour une explication des attributs de ce message, voir la description de la commande [DISPLAY CONN](#) .

Récupération des unités de récupération en attente de validation

Pour récupérer les unités de récupération en attente de validation, exécutez la commande suivante:

```
+CSQ1 RESOLVE INDOUBT( connection-name ) ACTION(COMMIT|BACKOUT)
NID( net-node.number )
```

où :

connection-name

ID système IMS .

ACTION

Indique si cette unité de récupération doit être validée (COMMIT) ou invalidée (BACKOUT).

net-node.number

net-node.numberassocié.

Lorsque vous avez émis la commande RESOLVE INDOUBT, l'un des messages suivants s'affiche:

```
CSQV414I +CSQ1 THREAD network-id COMMIT SCHEDULED
CSQV415I +CSQ1 THREAD network-id BACKOUT SCHEDULED
```

Résolution des postes de récupération résiduels

A des moments donnés, IMS génère une liste d'entrées de récupération résiduelles (RRE). Les REs sont des unités de récupération sur lesquelles IBM MQ peut être en attente de validation. Ils se produisent dans plusieurs situations:

- Si le gestionnaire de files d'attente n'est pas actif, IMS possède des RRE qui ne peuvent pas être résolues tant que le gestionnaire de files d'attente n'est pas actif. Ces REs ne sont pas un problème.
- Si le gestionnaire de files d'attente est actif et connecté à IMS, et si IMS annule le travail validé par IBM MQ, l'adaptateur IMS émet le message CSQQ010E. Si les données des deux systèmes doivent être cohérentes, il y a un problème. Pour plus d'informations sur la résolution de ce problème, voir «Récupération manuelle des unités de récupération IMS», à la page 443.
- Si le gestionnaire de files d'attente est actif et connecté à IMS, il se peut qu'il y ait encore des environnements d'exécution de récupération (RRE) même si aucun message ne vous a informé de ce problème. Une fois que la connexion IBM MQ à IMS a été établie, vous pouvez exécuter la commande IMS suivante pour déterminer s'il y a un problème:

```
/DISPLAY OASN SUBSYS sysid
```

Pour purger le RRE, exécutez l'une des commandes IMS suivantes:

```
/CHANGE SUBSYS sysid RESET  
/CHANGE SUBSYS sysid RESET OASN nnnn
```

où *nnnn* est le numéro de séquence d'application d'origine indiqué en réponse à la commande +CSQ1 DISPLAY. Il s'agit du numéro de planification de l'instance de programme, indiquant sa place dans la séquence d'appels de ce programme depuis le dernier démarrage à froid de IMS. IMS ne peut pas avoir deux unités de récupération en attente de validation avec le même numéro de planification.

Ces commandes réinitialisent le statut de IMS ; elles n'entraînent aucune communication avec IBM MQ.

Contrôle des connexions de région dépendante IMS

Vous pouvez contrôler, surveiller et, si nécessaire, arrêter les connexions entre IMS et IBM MQ.

Le contrôle des connexions de région dépendante IMS implique les activités suivantes:

- [Connexion à partir de régions dépendantes](#)
- [Options d'erreur de région](#)
- [Surveillance de l'activité sur les connexions](#)
- [Déconnexion des régions dépendantes](#)

Connexion à partir de régions dépendantes

L'adaptateur IMS utilisé dans la région de contrôle est également chargé dans les régions dépendantes. Une connexion est établie entre chaque région dépendante et IBM MQ. Cette connexion est utilisée pour coordonner l'engagement des travaux IBM MQ et IMS. Pour initialiser et établir la connexion, IMS effectue les opérations suivantes:

1. Lit le membre de sous-système (SSM) à partir de IMS.PROCLIB.

Un membre de sous-système peut être indiqué dans le paramètre EXEC de la région dépendante. S'il n'est pas indiqué, le SSM de la région de contrôle est utilisé. Si la région n'est jamais susceptible de se connecter à IBM MQ, pour éviter de charger l'adaptateur, spécifiez un membre sans entrée.

2. Charge l'adaptateur IBM MQ .

Pour un programme de message par lots, le chargement n'est pas effectué tant que l'application n'émet pas sa première commande de messagerie. A ce moment-là, IMS tente d'établir la connexion.

Pour une région de programme de traitement des messages ou une région de raccourci IMS , la tentative est effectuée lors de l'initialisation de la région.

Options d'erreur de région

Si le gestionnaire de files d'attente n'est pas actif ou si des ressources ne sont pas disponibles lorsque la première commande de messagerie est envoyée à partir de programmes d'application, l'action effectuée dépend de l'option d'erreur indiquée sur l'entrée SSM. Options :

R

Le code retour approprié est envoyé à l'application.

Q

L'application se termine de manière anormale avec le code de fin anormale U3051. Le message d'entrée est replacé en file d'attente.

A

L'application se termine de manière anormale avec le code de fin anormale U3047. Le message d'entrée est supprimé.

Surveillance de l'activité sur les connexions

Une unité d'exécution est établie à partir d'une région dépendante lorsqu'une application effectue sa première demande IBM MQ réussie. Vous pouvez afficher des informations sur les connexions et les applications qui les utilisent actuellement en exécutant la commande suivante à partir de IBM MQ:

```
+CSQ1 DISPLAY CONN(*) ALL
```

La commande génère un message similaire au suivant:

```

CONN(BC45A794C4290001)
EXTCONN(C3E2D8C3C3E2D8F140404040404040)
TYPE(CONN)
CONNOPTS(
MQCNO_STANDARD_BINDING
)
UOWLOGDA(2004-12-15)
UOWLOGTI(16.39.43)
UOWSTDA(2004-12-15)
UOWSTTI(16.39.43)
UOWSTATE(ACTIVE)
NID( )
EXTURID(
0000052900000000
)
QMURID(00000354B76E)
URTYPE(IMS)
USERID(STCPI)
APPLTAG(IM8F)
ASID(0049)
APPLTYPE(IMS)
PSTID(0004)
PSBNAME(GP01MPP)

```

Pour la région de contrôle, *thread-xref* est la valeur spéciale CONTROL. Pour les régions dépendantes, il s'agit du PSTid concaténé avec le PSBname. *auth-id* est soit la zone utilisateur de la carte de travail, soit l'ID de la table des procédures démarrées z/OS .

Pour une explication de la liste affichée, voir la description du message CSQV402I dans la documentation [Messages, codes achèvement et codes anomalie IBM MQ for z/OS](#) .

IMS fournit une commande d'affichage pour surveiller la connexion à IBM MQ. Il indique quel programme est actif sur chaque connexion de région dépendante, le nom d'utilisateur LTERM et l'état de la connexion de région de contrôle. La commande est la suivante :

```
/DISPLAY SUBSYS name
```

Le statut de la connexion entre IMS et IBM MQ est l'un des suivants:

```

CONNECTED
NOT CONNECTED
CONNECT IN PROGRESS
STOPPED
STOP IN PROGRESS
INVALID SUBSYSTEM NAME= name
SUBSYSTEM name NOT DEFINED BUT RECOVERY OUTSTANDING

```

Le statut de l'unité d'exécution de chaque région dépendante est l'un des suivants:

```

CONN
CONN, ACTIVE (includes LTERM of user)

```

Déconnexion des régions dépendantes

Permet de modifier les valeurs du membre SSM de IMS.PROCLIB, vous déconnectez une région dépendante. Pour ce faire, vous devez exécuter les opérations suivantes :

1. Exécutez la commande IMS :

```
/STOP REGION
```

2. Mettez à jour le membre SSM.
3. Exécutez la commande IMS :

```
/START REGION
```

déconnexion deIMS

La connexion est arrêtée lorsque IMS ou le gestionnaire de files d'attente s'arrête. L'opérateur de terminal maître IMS peut également interrompre explicitement la connexion.

Pour mettre fin à la connexion entre IMS et IBM MQ, utilisez la commande IMS suivante:

```
/STOP SUBSYS sysid
```

La commande envoie le message suivant au terminal qui l'a émis, généralement l'opérateur de terminal maître (MTO):

```
DFS058I STOP COMMAND IN PROGRESS
```

La commande IMS :

```
/START SUBSYS sysid
```

est nécessaire pour rétablir la connexion.

Remarque : La IMS commande /STOP SUBSYS n'est pas exécutée si un moniteur de déclenchement IMS est en cours d'exécution.

Contrôle du moniteur de déclenchement IMS

Vous pouvez utiliser la transaction CSQQTRMN pour arrêter et démarrer le moniteur de déclenchement IMS .

Le moniteur de déclenchement IMS (transaction CSQQTRMN) est décrit dans la rubrique [Configuration du moniteur de déclenchement IMS](#).

Pour contrôler le moniteur de déclenchement IMS , voir:

- [Démarrage de CSQQTRMN](#)
- [Arrêt de CSQQTRMN](#)

Démarrage de CSQQTRMN

1. Démarrez une BMP orientée par lots qui exécute le programme CSQQTRMN pour chaque file d'attente d'initialisation à surveiller.
2. Modifiez votre JCL par lots pour ajouter un nom symbolique de CSQQUT1 qui pointe vers un fichier contenant les informations suivantes:

```
QMGRNAME=q_manager_name      Comment: queue manager name
INITQUEUENAME=init_q_name     Comment: initiation queue name
LTERM=lterm                   Comment: LTERM to remove error messages
CONSOLEMESSAGES=YES          Comment: Send error messages to console
```

où :

| | |
|----------------------|--|
| q_manager_name | Nom du gestionnaire de files d'attente (si cette zone est vide, la valeur par défaut nommée dans CSQQDEFV est supposée) |
| init_q_name | Nom de la file d'attente d'initialisation à surveiller |
| lterm | Nom LTERM IMS pour la destination des messages d'erreur (si cette zone est vide, la valeur par défaut est MASTER). |
| CONSOLEMESSAGES= YES | Demande que les messages envoyés au terminal logique IMS désigné soient également envoyés à la console z/OS . Si ce paramètre est omis ou mal orthographié, la valeur par défaut n'est PAS l'envoi de messages à la console. |

3. Ajoutez le nom de définition de données CSQQUT2 si vous souhaitez un rapport imprimé du traitement de l'entrée CSQQUT1 .

Remarque :

1. Le fichier CSQQUT1 est défini avec LRECL=80. Les autres informations du bloc de contrôle des données sont extraites du fichier. Le bloc de contrôle des données du fichier CSQQUT2 est RECFM=VBA et LRECL=125.
2. Vous ne pouvez placer qu'un seul mot clé sur chaque enregistrement. La valeur du mot clé est délimitée par le premier blanc qui suit le mot clé ; cela signifie que vous pouvez inclure des commentaires. Un astérisque dans la colonne 1 signifie que l'ensemble de l'enregistrement d'entrée est un commentaire.
3. Si vous avez mal orthographié l'un des mots clés QMGRNAME ou LTERM , CSQQTRMN utilise la valeur par défaut pour ce mot clé.
4. Vérifiez que le sous-système est démarré dans IMS (à l'aide de la commande /START SUBSYS) avant de soumettre le travail BMP du moniteur de déclenchement. S'il n'est pas démarré, votre travail de moniteur de déclenchement s'arrête avec le code de fin anormale U3042.

Arrêt de CSQQTRMN

Une fois démarré, CSQQTRMN s'exécute jusqu'à ce que la connexion entre IBM MQ et IMS soit interrompue en raison de l'un des événements suivants:

- arrêt du gestionnaire de files d'attente
- IMS arrêter

ou une commande z/OS STOP **jobname** est entrée.

Contrôle du pont IMS

Utilisez cette rubrique pour comprendre les commandes IMS que vous pouvez utiliser pour contrôler le pont IMS .

Il n'existe aucune commande IBM MQ pour contrôler le pont IBM MQ-IMS . Toutefois, vous pouvez arrêter la distribution des messages à IMS de l'une des manières suivantes:

- Pour les files d'attente non partagées, à l'aide de la commande ALTER QLOCAL (xxx) GET (DISABLED) pour toutes les files d'attente de pont.
- Pour les files d'attente en cluster, à l'aide de la commande SUSPEND QMGR CLUSTER (xxx). Cela ne s'applique que lorsqu'un autre gestionnaire de files d'attente héberge également la file d'attente de pont en cluster.
- Pour les files d'attente en cluster, à l'aide de la commande SUSPEND QMGR FACILITY (IMSBRIDGE). Aucun autre message n'est envoyé à IMS, mais les réponses des transactions en attente sont reçues de IMS.

Pour recommencer à envoyer des messages à IMS , exécutez la commande RESUME QMGR FACILITY (IMSBRIDGE).

Vous pouvez également utiliser la commande MQSC DISPLAY SYSTEM pour indiquer si le pont est suspendu.

Pour plus de détails sur ces commandes, voir [Commandes MQSC](#) .

Pour plus d'informations, voir:

- [«Démarrage et arrêt du pont IMS»](#), à la page 456
- [«Contrôle des connexions IMS»](#), à la page 456
- [Contrôle des files d'attente de pont](#)
- [«Resynchronisation du pont IMS»](#), à la page 458
- [Utilisation des noms tpipe](#)
- [Suppression de messages de IMS](#)
- [Suppression de tpipes](#)
- [«Expiration de la transaction IMS»](#), à la page 460

Démarrage et arrêt du pont IMS

Démarrez le pont IBM MQ en démarrant OTMA. Utilisez la commande IMS :

```
/START OTMA
```

ou démarrez-le automatiquement en spécifiant OTMA=YES dans les paramètres système IMS . Si OTMA est déjà démarré, le pont démarre automatiquement lorsque le démarrage du gestionnaire de files d'attente est terminé. Un message d'événement IBM MQ est généré lorsque OTMA est démarré.

Utilisez la commande IMS :

```
/STOP OTMA
```

pour arrêter la communication OTMA. Lorsque cette commande est émise, un message d'événement IBM MQ est généré.

Contrôle des connexions IMS

IMS fournit les commandes d'opérateur suivantes pour contrôler et surveiller la connexion à IBM MQ:

/DEQUEUE TMEMBER *tmember* TPIPE *tpipe*

Supprime des messages d'une ressource Tpipe. Indiquez PURGE pour supprimer tous les messages ou PURGE1 pour supprimer le premier message uniquement.

/DISPLAY OTMA

Affiche des informations récapitulatives sur le serveur et les clients OTMA, ainsi que sur le statut du client.

/DISPLAY TMEMBER *nom*

Affiche des informations sur un client OTMA.

/DISPLAY TRACE TMEMBER *nom*

Affiche des informations sur les éléments tracés.

/SECURE OTMA

Définit les options de sécurité.

/START OTMA

Active les communications via OTMA.

/START TMEMBER *tmember* TPIPE *tpipe*

Démarre la ressource Tpipe nommée.

/ARRETER OTMA

Arrête les communications via OTMA.

/STOP TMEMBER *tmember* TPIPE *tpipe*

Arrête la ressource Tpipe nommée.

/TRACE

Contrôle la trace IMS .

Pour plus d'informations sur ces commandes, voir le manuel *IMS/ESA Operators Reference* pour le niveau de IMS que vous utilisez.

Les réponses de commande IMS sont envoyées au terminal à partir duquel la commande a été émise. L'autorisation d'émettre des commandes IMS est basée sur la sécurité IMS .

Contrôle des files d'attente de pont

Pour arrêter la communication avec le gestionnaire de files d'attente avec le nom de membre XCF *tmember* via le pont, exécutez la commande IMS suivante:

```
/STOP TMEMBER tmember TPIPE ALL
```

Pour reprendre la communication, exécutez la commande IMS suivante:

```
/START TMEMBER tmember TPIPE ALL
```

Les tubes d'une file d'attente peuvent être affichés à l'aide de la commande MQ DISPLAY QUEUE.

Pour arrêter la communication avec le gestionnaire de files d'attente sur un seul Tpipe, exécutez la commande IMS suivante:

```
/STOP TMEMBER tmember TPIPE tpipe
```

Un ou deux Tpipes sont créés pour chaque file d'attente de pont active. Par conséquent, l'émission de cette commande arrête la communication avec la file d'attente IBM MQ . Pour reprendre la communication, utilisez la commande IMS suivante:

```
/START TMEMBER tmember TPIPE tpipe
```

Vous pouvez également modifier les attributs de la file d'attente IBM MQ pour empêcher son extraction.

Resynchronisation du pont IMS

Le pont IMS est automatiquement redémarré chaque fois que le gestionnaire de files d'attente, IMSou OTMA sont redémarrés.

La première tâche effectuée par le pont IMS consiste à resynchroniser avec IMS. Cela implique la vérification des numéros de séquence IBM MQ et IMS sur chaque ressource Tpipe synchronisée. Une ressource Tpipe synchronisée est utilisée lorsque des messages persistants sont envoyés à IMS à partir d'une file d'attente de pont IBM MQ - IMS à l'aide du mode de validation zéro (commit-then-send).

Si le pont ne peut pas se resynchroniser avec IMS, le code de détection IMS est renvoyé dans le message CSQ2023E et la connexion à OTMA est arrêtée. Si le pont ne peut pas se resynchroniser avec une ressource Tpipe IMS individuelle, le code de détection IMS est renvoyé dans le message CSQ2025E et la ressource Tpipe est arrêtée. Si une ressource Tpipe a été démarrée à froid, les numéros de séquence récupérables sont automatiquement réinitialisés à 1.

Si le pont détecte des numéros de séquence non concordants lors de la resynchronisation avec une ressource Tpipe, le message CSQ2020E est émis. Utilisez la commande IBM MQ RESET TPIPE pour lancer la resynchronisation avec IMS Tpipe. Vous devez fournir le nom du groupe et du membre XCF, ainsi que le nom de la ressource Tpipe ; ces informations sont fournies par le message.

Vous pouvez également indiquer :

- Nouveau numéro de séquence récupérable à définir dans la ressource Tpipe pour les messages envoyés par IBM MQet à définir comme numéro de séquence de réception du partenaire. Si vous ne le spécifiez pas, le numéro de séquence de réception du partenaire est défini sur le numéro de séquence d'envoi IBM MQ en cours.
- Un nouveau numéro de séquence récupérable à définir dans la ressource Tpipe pour les messages reçus par IBM MQet à définir comme numéro de séquence d'envoi du partenaire. Si vous ne le spécifiez pas, le numéro de séquence d'envoi du partenaire est défini sur le numéro de séquence de réception IBM MQ en cours.

Si une unité de récupération non résolue est associée à la ressource Tpipe, elle est également notifiée dans le message. Utilisez la commande IBM MQ RESET TPIPE pour indiquer si l'unité de récupération doit être validée ou non. Si vous validez l'unité de récupération, le lot de messages a déjà été envoyé à IMSet est supprimé de la file d'attente de pont. Si vous revenez à l'unité de récupération, les messages sont renvoyés à la file d'attente du pont pour être envoyés ultérieurement à IMS.

Les canaux de communication Tpipes de mode de validation 1 (envoi puis validation) ne sont pas synchronisés.

Remarques relatives aux transactions de mode de validation 1

Dans IMS, les transactions de mode de validation 1 (CM1) envoient leurs réponses de sortie avant le point de synchronisation.

Une transaction CM1 peut ne pas être en mesure d'envoyer sa réponse, par exemple pour les raisons suivantes:

- La ressource Tpipe sur laquelle la réponse doit être envoyée est arrêtée
- OTMA est arrêté
- Le client OTMA (c'est-à-dire le gestionnaire de files d'attente) a disparu
- La file d'attente de réponse et la file d'attente de rebut ne sont pas disponibles

Pour ces raisons, l'application IMS qui envoie le message se termine de manière pseudo-anormale avec le code U0119. La transaction et le programme IMS ne sont pas arrêtés dans ce cas.

Ces raisons empêchent souvent l'envoi de messages dans IMS, ainsi que la distribution de réponses à partir de IMS. Une fin anormale U0119 peut se produire si:

- Le Tpipe, OTMA ou le gestionnaire de files d'attente est arrêté alors que le message se trouve dans IMS
- IMS répond sur une ressource Tpipe différente au message entrant et cette ressource Tpipe est arrêtée
- IMS répond à un client OTMA différent et ce client n'est pas disponible.

Chaque fois qu'une fin anormale U0119 se produit, le message entrant dans IMS et les messages de réponse dans IBM MQ sont perdus. Si la sortie d'une transaction CMO ne peut pas être distribuée pour l'une de ces raisons, elle est mise en file d'attente sur la ressource Tpipe dans IMS.

Utilisation des noms tpipe

La plupart des commandes utilisées pour contrôler le pont IBM MQ - IMS requièrent le nom *tpipe*. Utilisez cette rubrique pour comprendre comment vous pouvez trouver plus de détails sur le nom tpipe.

Vous avez besoin de noms *tpipe* pour la plupart des commandes qui contrôlent le pont IBM MQ - IMS. Vous pouvez obtenir les noms tpipe à partir de la commande DISPLAY QUEUE et noter les points suivants:

- Les noms tpipe sont affectés lorsqu'une file d'attente locale est définie
- une file d'attente locale reçoit deux noms tpipe, l'un pour la synchronisation et l'autre pour la non-synchronisation
- Les noms de ressource TPIPE ne seront connus de IMS qu'après une communication entre IMS et IBM MQ spécifique à cette file d'attente locale particulière.
- Pour qu'une ressource TPIPE puisse être utilisée par le pont IBM MQ - IMS, sa file d'attente associée doit être affectée à une classe de stockage dont les zones de nom de membre et de groupe XCF correctes sont renseignées

suppression des messages dansIMS

Un message destiné à IBM MQ via le pont IMS peut être supprimé si le Tmember / Tpipe est arrêté. Pour supprimer un message pour le gestionnaire de files d'attente dont le nom de membre XCF est *tmember*, exécutez la commande IMS suivante:

```
/DEQUEUE TMEMBER tmember TPIPE tpipe PURGE1
```

Pour supprimer tous les messages sur Tpipe, exécutez la commande IMS suivante:

```
/DEQUEUE TMEMBER tmember TPIPE tpipe PURGE
```

Suppression de tpipes

Vous ne pouvez pas supprimer IMS tpipes vous-même. Ils sont supprimés par IMS aux moments suivants:

- Les tpipes synchronisées sont supprimées lorsque IMS est démarré à froid.
- Les tpipes non synchronisées sont supprimées lorsque IMS est redémarré.

Expiration de la transaction IMS

Un délai d'expiration est associé à une transaction ; tout message IBM MQ peut être associé à un délai d'expiration. L'intervalle d'expiration est transmis de l'application à IBM MQ à l'aide de MQMD.Expiry . L'heure correspond à la durée d'un message avant son expiration, exprimée sous la forme d'une valeur en dixièmes de seconde. Une tentative d'exécution de la commande MQGET d'un message, postérieure à son expiration, entraîne la suppression du message de la file d'attente et le traitement de l'expiration. Le délai d'expiration diminue sous forme de flux de messages entre les gestionnaires de files d'attente sur un réseau IBM MQ . Lorsqu'un message IMS est transmis via le pont IMS à OTMA, le délai d'expiration du message restant est transmis à OTMA en tant que délai d'expiration de transaction.

Si un délai d'expiration est spécifié pour une transaction, OTMA fait expirer les transactions d'entrée à trois emplacements différents dans IMS:

- réception d'un message d'entrée à partir de XCF
- temps de mise en file d'attente des messages d'entrée
- temps GU de l'application

Aucune expiration n'est effectuée après l'heure GU.

La transaction EXPRTIME peut être fournie par:

- IMS définition de transaction
- En-tête de message OTMA IMS
- Exit utilisateur IMS DFSINSX0
- Commandes IMS CREATE ou UPDATE TRAN

IMS indique qu'une transaction est arrivée à expiration en mettant fin à une transaction avec 0243 et en émettant un message. Le message émis est soit DFS555I dans l'environnement de files d'attente non partagées, soit DFS2224I dans l'environnement de files d'attente partagées.

z/OS

Exploitation de Advanced Message Security sous z/OS

L'espace adresse Advanced Message Security accepte les commandes à l'aide de la commande z/OS MODIFY.

Pour entrer des commandes pour l'espace adresse Advanced Message Security , utilisez la commande z/OS MODIFY.

Par exemple :

```
F qmgr AMSM, cmd
```

Les commandes MODIFY suivantes sont acceptées:

| Commande | Option | Description |
|-----------|------------------------------|--|
| affichage | | Afficher les informations de version |
| REGENERER | keyRing politique TOUT | Actualisez les certificats de fichier de clés et / ou les politiques de sécurité. |
| SMFAUDIT | REUSSITE ECHEC TOUT | Indiquez si l'audit SMF est requis lorsqu'AMS protège / déprotège correctement les messages, lorsqu'AMS ne parvient pas à protéger / déprotéger les messages, ou les deux. |

Tableau 27. Commandes MODIFY de l'espace adresse Advanced Message Security (suite)

| Commande | Option | Description |
|----------|---------|---|
| SMFTYPE | 0 - 255 | Définissez le type d'enregistrement SMF à générer lorsque AMS protège / déprotège les messages. Pour désactiver l'audit SMF, indiquez le type d'enregistrement 0. |

Remarque : Pour spécifier une option, vous devez la séparer par une virgule. Exemple :

```
F qmgIAMS,REFRESH KEYRING
F qmgIAMS,SMFAUDIT ALL
F qmgIAMS,SMFTYPE 180
```

Commande REFRESH.

Une application émettant un appel MQOPEN va récupérer les modifications. Les applications existantes continuent d'utiliser les options à partir du moment où elles ont ouvert la file d'attente. Pour récupérer les modifications, une application doit fermer et rouvrir la file d'attente.

V 9.0.1

z/OS

Fournisseur de services IBM MQ for z/OS pour z/OS

Connect

Le fournisseur de services IBM MQ for z/OS pour z/OS Connect (MQ Service Provider) traitent les demandes qui arrivent via z/OS Connect. Le MQ Service Provider permet aux applications compatibles avec REST d'interagir avec des actifs z/OS qui sont exposés dans des files d'attente et des rubriques IBM MQ for z/OS. Il n'est pas nécessaire de se soucier du codage requis pour utiliser la messagerie asynchrone.

Important : z/OS Connect EE version 3.0.21.0 et versions ultérieures, fournit une version améliorée de MQ Service Provider qui prend en charge les fichiers archive de service. Vous devez migrer vers cette version de z/OS Connect EE et utiliser le MQ Service Provider intégré, au lieu d'utiliser le fournisseur de services fourni avec le produit IBM MQ for z/OS .

Dans les informations z/OS Connect EE dans IBM Documentation, les scénarios de démarrage rapide pour MQ Service Provider dans z/OS Connect EE sont fournis sous [Scénarios de démarrage rapide](#), et des informations de référence détaillées sont fournies sous [Utilisation du fournisseur de services IBM MQ](#).

Cette section contient les informations suivantes:

Information associée

Vidéo: [IBM MQ Service Provider for z/OS Connect \(YouTube\)](#)

V 9.0.1

z/OS

IBM MQ for z/OS Service Provider for z/OS Connect-

Présentation

Présentation de IBM MQ for z/OS Service Provider for z/OS Connect, expliquant les principes utilisés et les instructions utilisées par le fournisseur de services.

Important : z/OS Connect EE version 3.0.21.0 et versions ultérieures, fournit une version améliorée de MQ Service Provider qui prend en charge les fichiers archive de service. Vous devez migrer vers cette version de z/OS Connect EE et utiliser le MQ Service Provider intégré, au lieu d'utiliser le fournisseur de services fourni avec le produit IBM MQ for z/OS .

Dans les informations z/OS Connect EE dans IBM Documentation, les scénarios de démarrage rapide pour MQ Service Provider dans z/OS Connect EE sont fournis sous [Scénarios de démarrage rapide](#), et des informations de référence détaillées sont fournies sous [Utilisation du fournisseur de services IBM MQ](#).

MQ Service Provider prend en charge deux versions de z/OS Connect:

IBM z/OS Connect version 1 (z/OS Connect V1)

Il s'agit d'un composant de WebSphere Liberty Profile (WLP) sur z/OS, disponible sans frais supplémentaires, qui expose les actifs z/OS en tant qu'interface REST et permet aux applications distantes, telles que celles qui s'exécutent sur un périphérique mobile, de leur envoyer des données formatées à l'aide de JSON.

Pour plus d'informations, voir [IBM z/OS Connect overview](#).

Important :

1. MQ Service Provider for IBM z/OS Connect V1 requiert l'adaptateur de ressources IBM WebSphere MQ 7.5 . La date de fin de service est le 30th avril 2018.

Après ce point, MQ Service Provider ne sera plus pris en charge avec IBM z/OS Connect V1 et vous devez utiliser IBM z/OS Connect EE à la place.

Pour cette raison, IBM z/OS Connect V1 est mieux placé pour le développement et la validation de concept, plutôt que dans les environnements de production.

2. MQ Service Provider prend uniquement en charge le code z/OS Connect V1 fourni dans WLP 8.5.5.9 et versions ultérieures.

IBM z/OS Connect Enterprise Edition (z/OS Connect EE)

Il s'agit d'un produit distinct, facturable, basé sur WLP, qui fournit toutes les fonctions dans IBM z/OS Connect V1, ainsi qu'un certain nombre d'améliorations, notamment des outils permettant de générer des API RESTful-l'éditeur d'API et l'intégration à IBM API Connect.

Pour plus d'informations, voir [IBM z/OS Connect EE](#).

Important : MQ Service Provider prend uniquement en charge z/OS Connect EE V2.0.3.0 (APAR [PI66869](#)) ou version ultérieure.

MQ Service Provider fournit la même fonction, quelle que soit la version de z/OS Connect utilisée. Toutes les fonctionnalités de z/OS Connect EE, telles que l'éditeur d'API, sont prises en charge avec MQ Service Provider.

Dans toute la documentation, le terme *z/OS Connect* est utilisé pour désigner les deux versions, sauf s'il est nécessaire de faire référence à une version spécifique, auquel cas l'un des noms définis dans le texte précédent est utilisé.

Les actifs z/OS exposés par z/OS Connect n'ont pas besoin de comprendre REST ou JSON, car z/OS Connect mappe les appels REST aux appels locaux et fournit également des transformations pour la conversion entre les structures de données JSON et locales, telles que les copybooks COBOL.

MQ Service Provider est disponible en tant que fonction WLP distincte qui peut être installée dans l'installation z/OS Connect appropriée. Notez que la majorité du travail de configuration de MQ Service Provider est basée sur la configuration, de sorte que l'application REST n'a pas besoin de connaître IBM MQ. Toutefois, l'accès à des éléments tels que des zones MQMD est fourni pour des applications plus avancées.

MQ Service Provider expose les files d'attente et les rubriques IBM MQ en tant que [«Services»](#), à la page 463.

Versions prises en charge de IBM MQ

MQ Service Provider est pris en charge avec IBM MQ for z/OS 8.0 et versions ultérieures.

Les versions antérieures à IBM MQ 9.0.1 ne sont pas fournies avec MQ Service Provider. A la place, voir [«Obtention de MQ Service Provider»](#), à la page 466 pour plus d'informations sur la façon d'obtenir le MQ Service Provider.

La documentation décrite dans les sections z/OS Connect de IBM Documentation s'applique également à toutes les versions prises en charge de IBM MQ.

Autres considérations

Le MQ Service Provider utilise les éléments suivants:

- HTTP GET pour un appel MQGET non destructif
- HTTP DELETE pour un appel MQGET destructif
- HTTP POST pour un appel MQPUT

Il s'agit des mêmes instructions utilisées par IBM MQ Bridge for HTTP. De même, MQ Service Provider utilise des en-têtes HTTP pour spécifier divers éléments, tels que des valeurs MQMD.

Pour plus d'informations sur les appels HTTP, voir [«Instructions prises en charge par MQ Service Provider»](#), à la page 464

Notez que les seules données représentées en tant que données JSON sont le contenu du message. Cela signifie que les applications ne sont pas inutilement exposées aux concepts IBM MQ si elles n'ont besoin que des données de message.

MQ Service Provider utilise la prise en charge de l'adaptateur de ressources IBM MQ dans WLP et, par conséquent, est basé sur JMS.

V 9.0.1 z/OS Services

MQ Service Provider expose les files d'attente et les rubriques IBM MQ, ainsi que les applications qui les sous-tendent, en tant que services. Il existe deux types de services: unidirectionnel et bidirectionnel qui sont décrits dans cette section.

Service unidirectionnel

Un service unidirectionnel peut être utilisé pour fournir une API RESTful sur une seule file d'attente ou rubrique IBM MQ. Les clients RESTful peuvent émettre une demande HTTP POST avec un contenu JSON vers un service unidirectionnel et le service prend le contenu et envoie un message, avec le contenu comme corps de message, à la file d'attente ou au sujet cible.

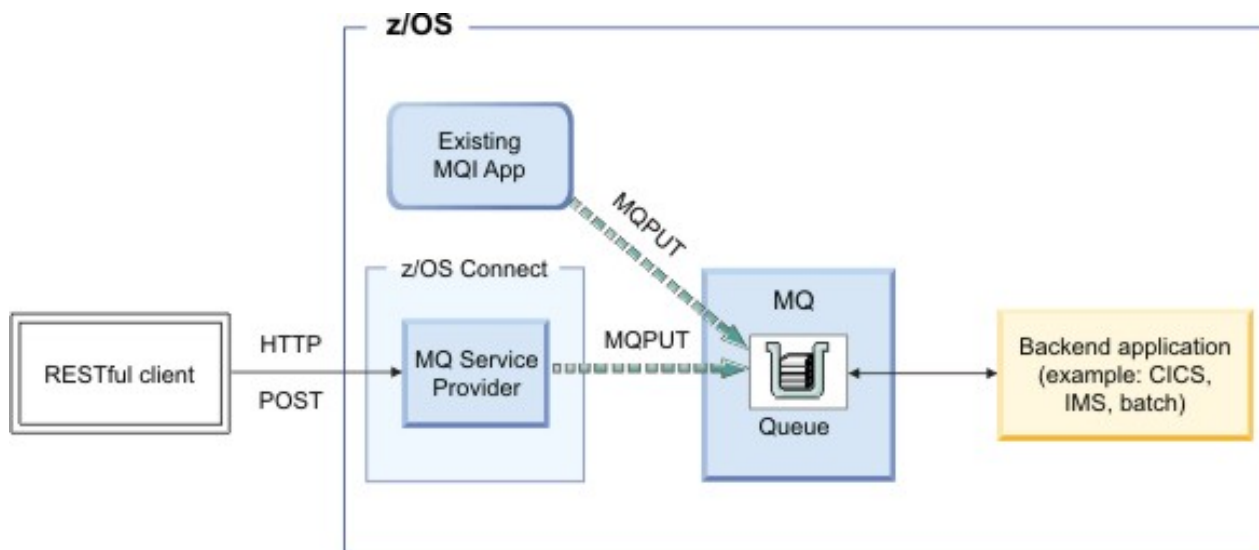


Figure 55. Service unidirectionnel

Les services unidirectionnels permettent également d'émettre des demandes HTTP DELETE et HTTP GET sur des files d'attente IBM MQ.

Une requête HTTP DELETE génère une extraction destructive d'un message disponible dans la file d'attente. Une requête HTTP GET permet de parcourir le premier message disponible dans la file d'attente.

Remarque : Deux appels HTTP GET renvoient le même message, sauf si une autre action s'est produite, par exemple une opération HTTP DELETE ou l'expiration d'un message, pour supprimer le message de la file d'attente.

Le corps du message est renvoyé au client RESTful sous la forme de JSON. Si le corps du message n'est pas déjà au format JSON, par exemple un copybook COBOL, z/OS Connect peut être configuré pour le convertir au format JSON, à l'aide d'une transformation de données.

Service bidirectionnelle

Un service bidirectionnel permet à un client RESTful d'exécuter une messagerie de type demande-réponse sur une paire de files d'attente.

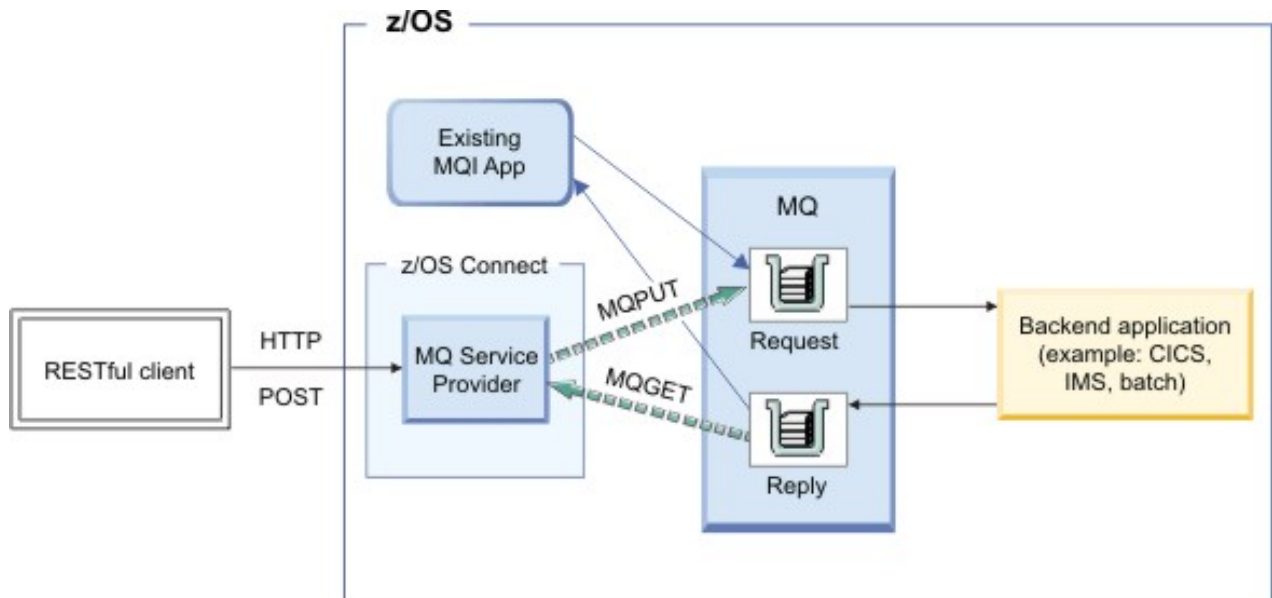


Figure 56. Service bidirectionnelle

Le client émet une demande HTTP POST spécifiant un contenu JSON. Le service prend le contenu, le convertit éventuellement dans un format différent, tel qu'un fichier de stockage COBOL, et l'envoie sous forme de message à une file d'attente de demandes.

Une application de back end consomme le message, le traite et génère une réponse qui est placée dans une file d'attente de réponses. Le service localise ce message, prend son contenu, le convertit éventuellement en JSON et le renvoie en tant que corps de réponse de l'autotest à la mise sous tension HTTP.

V 9.0.1 z/OS Instructions prises en charge par MQ Service Provider

MQ Service Provider prend en charge les instructions HTTP GET, HTTP DELETE et HTTP POST telles qu'utilisées dans IBM MQ Bridge for HTTP, avec la même signification. L'utilisation de ces verbes permet une interaction de base avec IBM MQ for z/OS, et la composition d'un certain nombre de ces verbes permet d'exposer des fonctions plus compliquées.

Lors d'une opération HTTP GET ou HTTP DELETE, MQ Service Provider ignore tout corps fourni dans la demande.

Si l'opération HTTP GET ou HTTP DELETE aboutit:

- La charge de message est renvoyée dans le corps de la réponse HTTP et le code de statut HTTP est 200.
- Un message sans contenu est renvoyé sans corps HTTP. Si aucun message n'est disponible, aucun corps HTTP n'est renvoyé et le code de statut HTTP est 204 (Aucun contenu).



Avertissement : z/OS Connect s'attend à ce que les contenus JSON soient toujours au format objet, c'est-à-dire que le contenu valide minimal soit `{ }`.

Lors d'une opération HTTP POST, les seuls paramètres attendus sont la charge de message, qui se trouve dans le corps HTTP, et les différents éléments décrits dans [mqzOSConnectService element](#).

Si l'autotest à la mise sous tension aboutit, procédez comme suit:

Service unidirectionnel

Renvoie un code HTTP 204 (No Content) avec un contenu vide.

Service bidirectionnelle

Renvoie le contenu du message de réponse au format JSON ou, si aucun message de réponse n'était disponible, un corps de réponse vide avec un code de réponse HTTP 204 (aucun contenu).

En-têtes HTTP

Lors de l'émission d'une demande HTTP POST vers un service unidirectionnel ou bidirectionnel, le corps HTTP doit être un objet JSON valide. Vous devez indiquer :

- Un en-tête HTTP Content-Type=application/json à HTTP_POST
- Codage UTF-8

Un certain nombre d'en-têtes HTTP facultatifs peuvent également être spécifiés, en fonction du verbe HTTP utilisé, pour ajuster le comportement de MQ Service Provider. Pour plus de détails, voir [En-têtes HTTP pouvant être utilisés avec MQ Service Provider](#) .

Codes HTTP

Si le MQ Service Provider détecte une erreur, le service définit un code de statut HTTP compris entre 400-599. Sinon, les codes de statut documentés précédemment sont toujours renvoyés.

Erreur côté client

Si des données non valides sont transmises par l'application appelante, un code de statut HTTP compris entre 400 et 499 est renvoyé à l'appelant. Le code de statut exact dépend de l'erreur.

Si les données non valides ont été détectées par une exception émise, une trace de pile au format JSON est fournie comme dans l'exemple côté serveur.

Erreur côté serveur

Si le MQ Service Provider rencontre une erreur inattendue, la trace de pile est transformée au format JSON et renvoyée à l'appelant avec un code de statut HTTP de 500 (Erreur de serveur interne). Les diagnostics appropriés seront également consignés dans les journaux z/OS Connect.

Voici un exemple simplifié de contenu de réponse d'erreur:

```
{
  "errorText": "CSQZ0006E: An unexpected JMSEException occurred while processing a request
for the 'mq7' service. ",
  "exceptionData": {
    "exceptions": [
      {
        "stackTrace": [
          "com.ibm.msg.client.jms.DetailedInvalidDestinationException: JMSWMQ2008:
Failed to open MQ queue 'ThisQueueDoesNotExist'.",
          <further content removed for brevity>
        ]
      },
      {
        "stackTrace": [
          "com.ibm.mq.MQException: JMSCMQ0001: WebSphere MQ call failed with
compcode '2' ('MQCC_FAILED') reason '2085' ('MQRC_UNKNOWN_OBJECT_NAME').",
          <further content removed for brevity>
        ]
      }
    ]
  }
}
```

Le MQ Service Provider est pris en charge pour une utilisation avec IBM MQ for z/OS 8.0 et les versions ultérieures. Il existe deux façons d'obtenir le MQ Service Provider à utiliser avec n'importe quelle version prise en charge d' IBM MQ.

1. «[Obtention de MQ Service Provider à partir de la fonction Unix Systems Services Components de IBM MQ 9.0.1 \(ou version ultérieure\)](#)», à la page 466
2. «[Obtention du MQ Service Provider à partir de Fix Central](#)», à la page 466

Obtention de MQ Service Provider à partir de la fonction Unix Systems Services Components de IBM MQ 9.0.1 (ou version ultérieure)

Installez la fonction en suivant les instructions du répertoire du programme. MQ Service Provider se trouve dans le répertoire *PATHPREFIX/zosconnect* de votre système de fichiers, où *PATHPREFIX* est la valeur choisie lors de l'installation. Par défaut, pour IBM MQ 9.0.1, *PATHPREFIX* est */usr/lpp/mqm/V9R0M1*.

Obtention du MQ Service Provider à partir de Fix Central

Accédez à [Fix Central](#) et localisez MQ Service Provider en recherchant *IBM-MQ-zOSConnect-ServiceProvider*.

Vous devez localiser un fichier tar, par exemple, *9.0.1-IBM-MQ-zOSConnect-ServiceProvider.tar*, qui peut être téléchargé sur votre poste de travail.



Avertissement : Le même fichier tar est également utilisé avec IBM MQ 8.0 . Il n'existe pas de fichier tar distinct pour IBM MQ 8.0.

Une fois téléchargé, transférez le fichier tar dans un répertoire *PATHPREFIX* sur une partition logique z/OS appropriée. Ce répertoire doit être accessible à z/OS Connect.

Extrayez les fichiers en exécutant la commande suivante:

```
tar -oxvf 9.0.1-IBM-MQ-zOSConnect-ServiceProvider.tar
```

Modifiez les droits de la structure de répertoires en fonction de votre entreprise.

Structure de répertoires

Les deux méthodes d'obtention du MQ Service Provider permettent de créer la même structure de répertoire. Le répertoire racine de cette structure est appelé *MQSP_ROOT* dans cette documentation, où *MQSP_ROOT* correspond à *PATHPREFIX/zosconnect*.

Sous *MQSP_ROOT* se trouvent les fichiers et répertoires suivants:

| | |
|---------------------------------|--|
| mqzosconnect.properties | Properties file that can be copied into z/OS Connect |
| v1.0/ Connect V1 | Directory containing MQ Service Provider for z/OS |
| lib/ | |
| com.ibm.mq.zosconnect_1.0.0.jar | Feature jar file for MQ Service Provider for z/OS |
| Connect V1 | |
| features/ | |
| zosConnectMQ-1.0.mf | Feature manifest for MQ Service Provider for z/OS |
| Connect V1 | |
| v2.0/ Connect EE | Directory containing MQ Service Provider for z/OS |
| lib/ | |
| com.ibm.mq.zosconnect_2.0.0.jar | Feature jar file for MQ Service Provider for z/OS |
| Connect EE | |
| features/ | |

V 9.0.1

z/OS

Considérations liées aux transactions

HTTP n'étant pas un protocole transactionnel, aucune coordination transactionnelle des opérations de messagerie effectuées par MQ Service Provider n'est possible.

Dans ce cas, les implications sont les suivantes :

- Si une demande HTTP POST est envoyée à un service unidirectionnel et que la connexion échoue avant qu'une réponse HTTP ne soit reçue par le client, le client ne peut pas déterminer immédiatement si le message a été envoyé à la file d'attente ou à la rubrique configurée ou non.
- Si une requête HTTP DELETE est envoyée à un service unidirectionnel et que la connexion échoue avant qu'une réponse HTTP ne soit reçue par le client, il se peut qu'un message ait été extrait de la file d'attente de façon destructive et qu'il ait été perdu, car il n'est pas possible de réexécuter la requête destructive.
- Si une demande HTTP POST est envoyée à un service bidirectionnel et que la connexion échoue avant qu'une réponse HTTP ne soit reçue par le client, le client ne peut pas savoir où l'échec s'est produit. Il se peut que le message de demande ait été envoyé à la file d'attente des demandes ou que le message de réponse ait été reçu de la file d'attente des réponses et qu'il ait été perdu.
- Il n'est pas possible de coordonner le résultat de plusieurs instructions HTTP vers un service unidirectionnel ou bidirectionnel.

V 9.0.1

z/OS

IBM z/OS Connect EE -Procédure de configuration de MQ Service Provider

Une présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sous IBM z/OS Connect EE.

Pourquoi et quand exécuter cette tâche

Effectuez les tâches suivantes, dans l'ordre indiqué pour installer le MQ Service Provider et tous ses prérequis.

V 9.0.1

z/OS

Installation d' IBM z/OS Connect EE

Configuration requise et procédure d'installation d' IBM z/OS Connect EE.

Avant de commencer

Vérifiez que vous avez accès à [Update the z/OS Connect EE V2 embedded WLP](#).

Pourquoi et quand exécuter cette tâche

Cette procédure configure un serveur z/OS Connect EE pouvant être utilisé avec le MQ Service Provider. Si vous disposez déjà d'un serveur qui a été configuré de la même manière qu'il peut être utilisé à la place.

Procédure

1. Créez un serveur en suivant les instructions décrites ici: [Création d'un serveur z/OS Connect EE](#)
2. Activez le service autorisé TXRRS en suivant la procédure décrite dans [Configuration du processus ange Liberty et des services autorisés z/OS](#).
3. Vérifiez que le service autorisé TXRRS a été configuré correctement par:
 - a) Démarrer le serveur
Suivez la procédure décrite dans [Démarrage et arrêt de z/OS Connect EE](#)
 - b) Affichez les journaux du serveur qui se trouvent à l'emplacement suivant:

```
/var/zosconnect/servers/server_name/logs/messages.log
```

Il s'agit de fichiers ASCII qui doivent contenir une sortie similaire à la suivante:

```
A CWWKE0001I: The server test has been launched.
I CWWKB0103I: Authorized service group LOCALCOM is available.
I CWWKB0103I: Authorized service group PRODMGR is available.
I CWWKB0103I: Authorized service group SAFCREd is available.
I CWWKB0103I: Authorized service group TXRRS is available.
I CWWKB0103I: Authorized service group WOLA is available.
I CWWKB0103I: Authorized service group ZOSDUMP is available.
I CWWKB0103I: Authorized service group ZOSWLM is available.
I CWWKB0103I: Authorized service group CLIENT.WOLA is available.
I CWWKB0108I: IBM CORP product z/OS Connect version 02.00 successfully registered with z/OS
```

Vérifiez la sortie pour voir si le groupe de services autorisés TXRRS est disponible. Dans l'exemple précédent, la ligne en gras indique que le groupe de services est disponible.

Le MQ Service Provider a besoin que ce groupe de services soit disponible

c) Arrêt du serveur.

Suivez la procédure décrite dans [Démarrage et arrêt d'un serveur à partir de la ligne de commande](#)

Résultats

Vous avez correctement installé z/OS Connect EE.

Que faire ensuite

Installez l'adaptateur de ressources IBM MQ.

Tâches associées

«IBM z/OS Connect EE -Procédure de configuration de MQ Service Provider», à la page 467

Une présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sous IBM z/OS Connect EE.

«Installation de l'adaptateur de ressources IBM MQ», à la page 468

La fonction de fournisseur de messagerie IBM MQ dans IBM z/OS Connect EE utilise un composant de IBM MQ appelé adaptateur de ressources IBM MQ . L'adaptateur de ressources est fourni avec IBM MQ dans le cadre de la fonction IBM MQ for z/OS Unix System Services Components.

Installation de l'adaptateur de ressources IBM MQ

La fonction de fournisseur de messagerie IBM MQ dans IBM z/OS Connect EE utilise un composant de IBM MQ appelé adaptateur de ressources IBM MQ . L'adaptateur de ressources est fourni avec IBM MQ dans le cadre de la fonction IBM MQ for z/OS Unix System Services Components.

Avant de commencer

Assurez-vous que la fonction IBM MQ for z/OS Unix System Services Components, pour la version du gestionnaire de files d'attente à laquelle vous allez vous connecter, est installée.

Si vous prévoyez de vous connecter à plusieurs versions du gestionnaire de files d'attente, vous devez utiliser la fonction de la version la plus récente.

Pourquoi et quand exécuter cette tâche

Cette procédure installe l'adaptateur de ressources IBM MQ dans IBM z/OS Connect EE.

Procédure

1. Localisez le répertoire Unix System Services Components de la version de IBM MQ à laquelle vous allez vous connecter.

Par exemple, ce répertoire peut être appelé /usr/lpp/mqm/V9R0M1/. Le répertoire doit contenir un sous-répertoire java/lib, qui contient un certain nombre de bibliothèques natives (fichiers .so).



Avertissement : Si vous prévoyez de vous connecter à plusieurs versions du gestionnaire de files d'attente, utilisez la version la plus récente.

2. Editez le fichier server.xml du serveur z/OS Connect EE.

Ajoutez les lignes suivantes :

```
<variable name="wmqJmsClient.rar.location"
  value="MQJAVA_LIB_DIR/jca/wmq.jmsra.rar"/>
<wmqJmsClient nativeLibraryPath="MQJAVA_LIB_DIR"/>
```

où MQJAVA_LIB_DIR est basé sur le répertoire situé à l'étape «1», à la page 468, par exemple, /usr/lpp/mqm/V9R0M1/java/lib.

- b. Sauvegardez les modifications.

La première ligne indique à IBM z/OS Connect EE où localiser l'adaptateur de ressources IBM MQ.

La deuxième ligne indique à l'adaptateur de ressources IBM MQ où localiser les bibliothèques natives utilisées pour les connexions de liaisons à IBM MQ.

Pour plus d'informations, voir [ClientwmqJmsset Déploiement d'applications JMS dans Liberty pour utiliser le fournisseur de messagerie IBM MQ](#).

3. Configurez la bibliothèque STEPLIB de votre serveur pour inclure les bibliothèques IBM MQ.
Vous devez effectuer cette opération pour que les bibliothèques natives situées à l'étape «1», à la page 468 puissent se connecter au gestionnaire de files d'attente.

En règle générale, cela implique l'édition du JCL, qui démarre votre serveur, pour inclure les éléments suivants:

```
//STEPLIB DD DSN=HLQ.SCSQAUTH,DISP=SHR
// DD DSN=HLQ.SCSQANLE,DISP=SHR
```

où HLQ est le qualificatif de haut niveau des fichiers contenant l'installation IBM MQ.

Si vous prévoyez de vous connecter à plusieurs versions de IBM MQ à partir du même serveur, utilisez les fichiers de la version la plus récente.

Résultats

Vous avez partiellement installé l'adaptateur de ressources IBM MQ.

Notez que l'adaptateur de ressources IBM MQ ne sera pas entièrement installé tant que sa fonction parent (wmqJmsClient-2.0) ne sera pas activée, comme décrit dans [«Activation de z/OS Connect et de MQ Service Provider»](#), à la page 470.

Que faire ensuite

Installez MQ Service Provider dans IBM z/OS Connect EE.

Tâches associées

[«IBM z/OS Connect EE -Procédure de configuration de MQ Service Provider»](#), à la page 467

Une présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sous IBM z/OS Connect EE.

[«Installation d' IBM z/OS Connect EE»](#), à la page 467

Configuration requise et procédure d'installation d' IBM z/OS Connect EE.

[«Installation de MQ Service Provider dans IBM z/OS Connect EE»](#), à la page 470

Vous devez installer le MQ Service Provider dans IBM z/OS Connect EE avant de l'utiliser.

Vous devez installer le MQ Service Provider dans IBM z/OS Connect EE avant de l'utiliser.

Pourquoi et quand exécuter cette tâche

Pour ce faire, procédez comme suit:

Procédure

1. Localisez le répertoire des extensions de produit IBM z/OS Connect EE.

Pour une installation standard, il s'agit du répertoire `/var/zosconnect/v2r0/extensions` décrit ici: [Configuration du répertoire des extensions de produit](#).

2. Copiez `MQSP_ROOT/mqzosconnect.properties` dans le répertoire situé à l'étape «1», à la page [470](#).
3. Editez le fichier copié.

Notez que ce fichier est un fichier ASCII. Remplacez `PATH_TO_INSTALL` par `MQSP_ROOT/v2.0` et sauvegardez vos modifications.



Avertissement : L'étape «3», à la page [470](#) suppose que l'ID utilisateur exécutant le serveur dispose d'un accès en lecture à la structure de répertoire `MQSP_ROOT`. Si tel n'est pas le cas, ajoutez un accès suffisant à l'ID utilisateur ou déplacez le contenu de `MQSP_ROOT` vers un emplacement disposant d'un accès suffisant.

Résultats

Vous avez maintenant installé tous les prérequis pour MQ Service Provider.

Que faire ensuite

Vous devez maintenant activer MQ Service Provider et z/OS Connect.

Tâches associées

«[IBM z/OS Connect EE -Procédure de configuration de MQ Service Provider](#)», à la page [467](#)

Une présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sous IBM z/OS Connect EE.

«[Installation de l'adaptateur de ressources IBM MQ](#)», à la page [468](#)

La fonction de fournisseur de messagerie IBM MQ dans IBM z/OS Connect EE utilise un composant de IBM MQ appelé adaptateur de ressources IBM MQ. L'adaptateur de ressources est fourni avec IBM MQ dans le cadre de la fonction IBM MQ for z/OS Unix System Services Components.

«[Activation de z/OS Connect et de MQ Service Provider](#)», à la page [470](#)

Actions à effectuer pour activer z/OS Connect et MQ Service Provider.

Actions à effectuer pour activer z/OS Connect et MQ Service Provider.

Avant de commencer

Vérifiez que vous avez effectué les procédures suivantes:

- «[Installation d' IBM z/OS Connect EE](#)», à la page [467](#)
- «[Installation de l'adaptateur de ressources IBM MQ](#)», à la page [468](#)
- «[Installation de MQ Service Provider dans IBM z/OS Connect EE](#)», à la page [470](#)

Pourquoi et quand exécuter cette tâche

Cette procédure active à la fois z/OS Connect et MQ Service Provider.

Procédure

1. Editez le fichier `server.xml` du serveur z/OS Connect EE que vous avez créé et remplacez l'ensemble de l'élément **featureManager** par les lignes suivantes:

```
<featureManager>
  <feature>zosconnect:zosconnect-2.0</feature>
  <feature>appSecurity-2.0</feature>
  <feature>jms-2.0</feature>
  <feature>mqzosconnect:zosConnectMQ-2.0</feature>
  <feature>wmqJmsClient-2.0</feature>
  <feature>zosTransaction-1.0</feature>
</featureManager>
```



Avertissement : Modifiez ces entrées uniquement si elles n'existent pas déjà.

2. Configurez la sécurité pour z/OS Connect.
Pour plus d'informations sur l'exécution de cette procédure, voir [Configuration de la sécurité pour z/OS Connect EE](#).
3. Démarrer le serveur.

Que faire ensuite

Vérifiez que z/OS Connect est correctement configuré.

Tâches associées

«IBM z/OS Connect EE -Procédure de configuration de MQ Service Provider», à la [page 467](#)

Une présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sous IBM z/OS Connect EE.

«Vérification de la configuration correcte de z/OS Connect», à la [page 471](#)

Comment vérifier que z/OS Connect est correctement configuré.

Vérification de la configuration correcte de z/OS Connect

Comment vérifier que z/OS Connect est correctement configuré.

Avant de commencer

Vérifiez que vous avez effectué la procédure décrite dans «Activation de z/OS Connect et de MQ Service Provider», à la [page 470](#)

Pourquoi et quand exécuter cette tâche

z/OS Connect fournit une API RESTful qui peut être utilisée pour interroger les services installés et effectuer des actions d'administration telles que leur arrêt et leur démarrage.

Procédure

1. Emettez une requête HTTP GET sur z/OS Connect pour interroger la liste des services actuellement installés.

Pour ce faire, utilisez un navigateur Web en entrant une URL du formulaire

```
https://HOST_NAME:HTTPS_PORT/zosConnect/services
```

où `HOST_NAME` et `HTTPS_PORT` sont les valeurs que vous avez entrées à l'étape «1», à la [page 467](#) de «Installation d' IBM z/OS Connect EE», à la [page 467](#).

Par exemple :

https://yourdomainname:12342/zosConnect/services

2. Entrez l'ID utilisateur et le mot de passe lorsque vous y êtes invité par le navigateur.
Il s'agit des valeurs que vous avez entrées dans l'élément user à l'étape «2», à la page 471 de «Activation de z/OS Connect et de MQ Service Provider», à la page 470.

Résultats

Il en résulte la réponse JSON suivante, qui indique que z/OS Connect est en cours d'exécution, mais qu'aucun service n'est installé. Notez que si un serveur z/OS Connect existant était utilisé et que des services étaient installés, ces services s'afficheraient.

```
-----  
{  
  "zosConnectServices":[  ]  
}
```

Que faire ensuite

Configurez un service IBM MQ unidirectionnelle simple sur z/OS Connect EE.

Tâches associées

«IBM z/OS Connect EE -Procédure de configuration de MQ Service Provider», à la page 467

Une présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sous IBM z/OS Connect EE.

Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect EE

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

Avant de commencer

Vérifiez que vous avez configuré tous les composants correctement, comme décrit dans Activation d' z/OS Connect et de MQ Service Provider et Vérification de la configuration correcte de z/OS Connect

Procédure

1. Créez une file d'attente appelée ONE_WAY_QUEUE sur le gestionnaire de files d'attente z/OS cible, à l'aide de MQSC ou de IBM MQ Explorer.
2. Définissez une fabrique de connexions de fournisseur de messagerie IBM MQ et une file d'attente.
Pour ce faire, ajoutez ce qui suit au bas de `server.xml`, du serveur z/OS Connect EE que vous avez créé, mais dans l'élément `server`.

```
-----  
<jmsConnectionFactory id="cf1" jndiName="jms/cf1" connectionManagerRef="ConMgr1">  
  <properties.wmqJms  
    transportType="BINDINGS"  
    queueManager="MQ21" />  
</jmsConnectionFactory>  
  
<connectionManager id="ConMgr1" maxPoolSize="5" />  
  
<jmsQueue id="q1" jndiName="jms/d1">  
  <properties.wmqJms  
    baseQueueName="ONE_WAY_QUEUE" />  
</jmsQueue>  
-----
```

Remarques :

- a. Remplacez la valeur de l'attribut **queueManager** par le nom de gestionnaire de files d'attente cible correct.
 - b. Un **transportType** de liaisons est utilisé. Cela signifie qu'une connexion intermémoire est utilisée pour communiquer avec le gestionnaire de files d'attente. Il s'agit du seul **transportType** pris en charge lors de l'utilisation de MQ Service Provider.
3. Définissez un service IBM MQ unidirectionnel simple en ajoutant ce qui suit à `server.xml`, du serveur z/OS Connect EE que vous avez créé, mais dans l'élément `server`.

```
-----  
<zosconnect_zosConnectService id="zosconnMQ1"  
    invokeURI="/oneWay"  
    serviceName="oneWay"  
    serviceRef="oneWay" />  
  
<mqzosconnect_mqzOSConnectService id="oneWay "  
    connectionFactory="jms/cf1"  
    destination="jms/d1"/>  
-----
```

L'élément `zosConnectService` définit un nouveau service pour z/OS Connect à l'aide du **serviceName** de `oneWay`. Les éléments suivants:

- L'attribut **invokeURI** facilite l'appel du service.
- L'attribut **serviceRef** doit correspondre à l'attribut ID d'un fournisseur de services z/OS Connect, qui, dans ce cas, est fourni par l'élément `mqzOSConnectService`.

L'élément `mqzOSConnectService` définit une instance de service unique fournie par MQ Service Provider.

Les attributs **connectionFactory** et **destination** indiquent à l'instance comment localiser la fabrique de connexions du fournisseur de messagerie IBM MQ et la file d'attente, respectivement.

Pour plus d'informations sur les attributs de cette structure, voir [ElémentmqzOSConnectService](#).

Résultats

Vous avez configuré un service unidirectionnel simple.

Que faire ensuite

Vous devez tester le service.

Tâches associées

«Test du service unidirectionnel sur z/OS Connect EE», à la page 473

Une série d'étapes garantissant que votre service unidirectionnel fonctionne.

  *Test du service unidirectionnel sur z/OS Connect EE*

Une série d'étapes garantissant que votre service unidirectionnel fonctionne.

Avant de commencer

Vérifiez que vous avez correctement effectué la procédure [«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect EE»](#), à la page 472.

Procédure

1. Vérifiez que z/OS Connect reconnaît le nouveau service.

Pour ce faire, réexécutez les étapes décrites dans la rubrique [Vérification de la configuration correcte d' z/OS Connect](#).

Etant donné qu'un service est maintenant défini, vous devriez voir une sortie similaire à la suivante:

```

-----
{
  "zosConnectServices": [
    {
      "serviceName": "oneWay",
      "serviceDescription": "DATA_UNAVAILABLE",
      "serviceProvider": "IBM MQ for z/OS service provider for IBM z/OS Connect EE V2.0",
      "serviceURL": "https://yourdomainname:12342/zosConnect/services/oneWay"
    }
  ]
}
-----

```

Remarque : **ServiceDescription** est renvoyé en tant que DATA_UNAVAILABLE car l'attribut **serviceDescription** n'a pas été spécifié sur l'élément zosConnectService .

2. Émettez une demande HTTP POST au service. Cela entraîne l'envoi d'un nouveau message à la file d'attente ONE_WAY_QUEUE avec le contenu spécifié sur l'autotest à la mise sous tension.

Un bon moyen d'effectuer ce type de test consiste à utiliser un plug-in tel que le [client REST avancé \(ARC\)](#) pour Google Chrome.

- a) Utilisez ARC pour émettre une demande HTTP POST sur l'URL suivante:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

- b) Spécifiez un code JSON valide pour le corps, par exemple:

```
{"name1" : "value1", "name2" : "value2"}
```

Vous devez spécifier deux en-têtes:

- i) Autorisation = Basic *ENCODED_USERID_PASSWORD*

où *ENCODED_USERID_PASSWORD* est un ID utilisateur et un mot de passe codés en base 64, comme décrit dans le [Schéma d'authentification de base](#).

- ii) Type de contenu = *application/json*

Le premier en-tête est utilisé pour la connexion à z/OS Connect et est une forme codée de l'ID utilisateur et du mot de passe utilisés dans [Enabling z/OS Connect et MQ Service Provider](#).

Le deuxième en-tête informe le serveur que le contenu est JSON, qui est la seule valeur prise en charge lors de l'émission d'appels HTTP POST à MQ Service Provider.

Si vous ne spécifiez pas cet en-tête, une erreur HTTP 415 indiquant un type de support non pris en charge est générée.

L'émission d'une demande HTTP POST vers un service unidirectionnel entraîne le renvoi d'un corps de réponse HTTP vide et d'un code de réponse HTTP 204 (aucun contenu).

ARC génère un message indiquant que la réponse ne contient pas de données.

Pour vérifier qu'un message a été inséré, utilisez un outil tel que l'explorateur IBM MQ pour parcourir le contenu de ONE_WAY_QUEUE. Il doit y avoir un message IBM MQ au format MQSTR dans la file d'attente avec le contenu JSON qui a été envoyé.

3. Utilisez ARC pour émettre une requête HTTP DELETE sur l'URL suivante:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

Cette opération extrait de façon destructive le message qui vient d'être envoyé au service unidirectionnel.

Cela doit fournir un corps de réponse contenant le contenu JSON qui a été envoyé à l'origine. Vous pouvez utiliser IBM MQ Explorer pour confirmer que le message a été obtenu de façon destructive.

Tâches associées

«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect EE», à la page [472](#)

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

V 9.0.1 z/OS Configuration d'un service IBM MQ bidirectionnelle simple sur z/OS Connect EE

Comment configurer un service IBM MQ bidirectionnel dans z/OS Connect EE, sur une paire existante de files d'attente IBM MQ appelées MQ_REQUEST_Q et MQ_REPLY_Q.

Avant de commencer

Vérifiez que vous avez configuré tous les composants correctement, comme décrit dans [«Activation de z/OS Connect et de MQ Service Provider»](#), à la page 481 et [«Vérification de la configuration correcte de z/OS Connect»](#), à la page 482.

Pourquoi et quand exécuter cette tâche

Cette opération est légèrement plus importante que la configuration d'un service unidirectionnel, car vous avez besoin d'une application de back end pour consommer les messages de demande envoyés à l'aide de z/OS Connect et générer des messages de réponse.

Cette tâche suppose que les files d'attente MQ_REQUEST_Q et MQ_REPLY_Q sont utilisées par une application d'arrière-plan existante, par exemple une transaction CICS ou IMS, qui reçoit un message de demande de MQ_REQUEST_Q et génère un message de réponse placé sur MQ_REPLY_Q.

Procédure

1. Définissez une fabrique de connexions de fournisseur de messagerie IBM MQ et deux files d'attente de fournisseur de messagerie IBM MQ sur le serveur z/OS Connect EE.

Les files d'attente du fournisseur de messagerie IBM MQ sont mappées aux files d'attente MQ_REQUEST_Q et MQ_REPLY_Q existantes. Pour ce faire, ajoutez ce qui suit au bas de `server.xml`, du serveur z/OS Connect EE que vous avez créé, mais dans l'élément `server`.

```
<jmsConnectionFactory id="cf2" jndiName="jms/cf2"
    connectionManagerRef="ConMgr2">
  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager id="ConMgr2" maxPoolSize="5"/>

<jmsQueue id="requestQueue" jndiName="jms/requestQueue">
  <properties.wmqJms baseQueueName="MQ_REQUEST_Q"/>
</jmsQueue>

<jmsQueue id="replyQueue" jndiName="jms/replyQueue">
  <properties.wmqJms baseQueueName="MQ_REPLY_Q"/>
</jmsQueue>
```

Remarques :

- a. Remplacez la valeur de l'attribut `queueManager` par le nom de gestionnaire de files d'attente cible correct.
 - b. Vous devrez peut-être effectuer une configuration supplémentaire de ces éléments, en fonction de la nature de l'application de back end. Pour plus d'informations, voir [«Configuration de l'MQ Service Provider»](#), à la page 487.
 - c. L'attribut `maxPoolSize` peut avoir besoin d'être ajusté en fonction du nombre de demandes simultanées. Pour plus d'informations sur cet attribut, voir [Configuration du regroupement de connexions pour les connexions JMS](#).
2. Définissez le service IBM MQ bidirectionnel en ajoutant ce qui suit à `server.xml`, du serveur z/OS Connect EE que vous avez créé, mais dans l'élément `server`.

```

-----
<zosconnect_zosConnectService id="zosconnMQ2"
    invokeURI="/twoWay"
    serviceName="twoWay"
    serviceRef="twoWay" />

<mqzosconnect_mqzOSConnectService id="twoWay"
    connectionFactory="jms/cf2"
    destination="jms/requestQueue"
    replyDestination="jms/replyQueue"
    waitInterval="10000" />
-----

```

Remarques :

- a. Vous devrez peut-être effectuer une configuration supplémentaire de ces éléments, en fonction de la nature de l'application de back end. Par exemple, vous pouvez avoir besoin d'une transformation de données. Pour plus d'informations, voir [«Configuration de l'MQ Service Provider»](#), à la page 487.
- b. Vous devrez peut-être ajuster la valeur de **waitInterval**. Pour plus d'informations, voir [waitInterval](#).
- c. Dans cet exemple, on suppose que les messages placés sur la destination de réponse, par l'application de back end, ont un ID de corrélation identique à l'ID de message sur la destination de demande.

Si ce n'est pas le cas, vous devez ajouter l'attribut **replySelection** à l'élément `mqzosconnect_mqzOSConnectService` et définir la valeur de manière appropriée.

Pour plus d'informations, voir [replySelection](#).

Résultats

Vous avez configuré un service bidirectionnel dans z/OS Connect EE, qui peut être utilisé avec une application de back end existante qui utilise une paire de files d'attente IBM MQ.

Que faire ensuite

Vous pouvez appeler le service bidirectionnel en émettant une commande HTTP POST sur l'URL `https://host_name:port_no/twoWay`.

Les étapes à suivre sont similaires à celles décrites dans [«Test du service unidirectionnel sur z/OS Connect EE»](#), à la page 473, sauf que seule l'instruction HTTP POST est prise en charge et que vous devez envoyer des données de contenu JSON appropriées à l'application de back end existante.

Tâches associées

[«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect EE»](#), à la page 472

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

Génération d'un fichier d'archive de service (SAR)

Avant de pouvoir créer une API avec z/OS Connect EE, vous devez générer un fichier SAR.

Pourquoi et quand exécuter cette tâche

Vous pouvez le faire avec IBM MQ, en utilisant le processus CICS-WOLA décrit dans les informations *z/OS Connect EE* dans IBM Documentation. Pour plus d'informations, voir [Générer une archive de service à partir d'un copybook CICS COBOL](#).

Important : Le processus décrit dans le texte suivant est facultatif. Si vous ne souhaitez pas utiliser les fonctions fournies par l'éditeur d'API, vous n'avez pas besoin du fichier SAR. Toutefois, vous souhaitez peut-être utiliser les schémas JSON et les fichiers de liens décrits dans le lien précédent, si vous souhaitez utiliser la prise en charge de la transformation de données par z/OS Connect EE.

Le processus décrit dans le lien précédent utilise l'utilitaire **BAQLS2JS** fourni avec z/OS Connect EE.

Cet utilitaire:

1. Génère des schémas JSON et des fichiers de liens qui peuvent être utilisés avec la fonction de transformation de données fournie dans z/OS Connect EE.

Si vous souhaitez configurer ces fichiers dans z/OS Connect EE, vous les définissez dans la configuration via l'élément `zosconnect_zosConnectDataXform`.

Pour plus d'informations, voir [«Utilisation des transformations de données avec MQ Service Provider»](#), à la page 491.

2. Génère un fichier SAR qui contient les schémas JSON de l'étape précédente et une référence à l'attribut **serviceName** d'une instance de service z/OS Connect EE particulière.

La référence est fournie à **BAQLS2JS** à l'aide du paramètre SERVICE-NAME.

Par exemple, si SERVICE-NAME=MQ1WayBackend est défini en tant qu'attribut sur **BAQLS2JS**, **BAQLS2JS** génère un fichier SAR qui peut être utilisé avec une instance de service IBM MQ configurée comme suit:

```
<zosconnect_zosConnectService id="MQ1WayService"
    serviceName="MQ1WayBackend"
    serviceRef="mq1way"
    invokeURI="/mq1way"
    dataXformRef="xformJSON2Byte" />

<mqzosconnect_mqZOSConnectService id="mq1way"
    connectionFactory="jms/cf1"
    destination="jms/oneWayRequestQ" />
```

Résultats

Une fois généré, le fichier SAR peut être importé dans l'éditeur d'API pour générer une API. Pour plus d'informations, voir [Création d'une API à l'aide de l'éditeur d'API z/OS Connect EE](#).

Une fois qu'une API a été construite dans l'éditeur d'API, une archive d'API (AAR) est générée, qui peut ensuite être déployée dans z/OS Connect EE, où elle s'exécute sur le service référencé par le nom de service attribué dans **BAQLS2JS**. Pour plus d'informations, voir [Déploiement d'une API sur le serveur z/OS Connect EE](#).

Tâches associées

[«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect EE»](#), à la page 472

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

[«Configuration d'un service IBM MQ bidirectionnelle simple sur z/OS Connect EE»](#), à la page 475

Comment configurer un service IBM MQ bidirectionnel dans z/OS Connect EE, sur une paire existante de files d'attente IBM MQ appelées MQ_REQUEST_Q et MQ_REPLY_Q.

IBM z/OS Connect V1 -Procédure de configuration MQ Service Provider

Présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sur IBM z/OS Connect V1.

Pourquoi et quand exécuter cette tâche

Effectuez les tâches suivantes, dans l'ordre indiqué pour installer le MQ Service Provider et tous ses prérequis.

InstallationWebSphere Application Server Liberty

Configuration requise et procédure d'installation de WebSphere Application Server Liberty (WLP).

Avant de commencer

Vérifiez que vous avez accès à une version de WLP for z/OS qui inclut z/OS Connect, par exemple 16.0.0.2.

Pourquoi et quand exécuter cette tâche

Cette procédure installe WLP et configure un serveur.

Procédure

1. Installez WLP en suivant la procédure décrite dans [Installation de Liberty à l'aide d' Installation Manager](#).
Notez que pour le reste de ces instructions, cette structure de répertoire est appelée *WLP_ROOT*.
2. Créez un nouveau serveur en suivant la procédure décrite dans [Création manuelle d'un serveur Liberty](#)
Voir [Configuration de z/OS Connect](#) qui décrit comment activer z/OS Connect sur le serveur créé.



Avertissement : Vous n'avez pas besoin d'effectuer les étapes facultatives *WebSphere Optimized Local Adapters (WOLA)* .

3. Activez le service autorisé TXRRS en suivant la procédure décrite dans [Activation des services autorisés z/OS sur Liberty for z/OS](#)
4. Vérifiez que le service autorisé TXRRS a été configuré correctement par:
 - a) Démarrer le serveur
Suivez la procédure décrite dans [Démarrage et arrêt d'un serveur à partir de la ligne de commande](#)
 - b) Affichez les journaux du serveur qui se trouvent à l'emplacement suivant:

```
WLP_ROOT/usr/servers/server_name/logs/message.log
```

Il s'agit de fichiers ASCII qui doivent contenir une sortie similaire à la suivante:

```
A CWWKE0001I: The server server1 has been launched.
I CWWKB0103I: Authorized service group LOCALCOM is available.
I CWWKB0103I: Authorized service group PRODMGR is available.
I CWWKB0103I: Authorized service group SAFCRE is available.
I CWWKB0103I: Authorized service group TXRRS is available.
I CWWKB0103I: Authorized service group WOLA is available.
I CWWKB0103I: Authorized service group ZOSDUMP is available.
I CWWKB0103I: Authorized service group ZOSWLM is available.
I CWWKB0103I: Authorized service group CLIENT.WOLA is available.
I CWWKB0108I: IBM CORP product WAS FOR Z/OS version 16.0 successfully registered with z/OS.
```

Vérifiez la sortie pour voir si le groupe de services autorisés TXRRS est disponible. Dans l'exemple précédent, la ligne en gras indique que le groupe de services est disponible.

MQ Service Provider a besoin que ce groupe de services soit disponible.

- c) Arrêt du serveur.
Suivez la procédure décrite dans [Démarrage et arrêt d'un serveur à partir de la ligne de commande](#)

Résultats

L'installation de WLP a abouti.

Que faire ensuite

Installez l'adaptateur de ressources IBM MQ.

Tâches associées

«IBM z/OS Connect V1 -Procédure de configuration MQ Service Provider», à la page 477

Présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sur IBM z/OS Connect V1 .

«Installation de l'adaptateur de ressources IBM MQ», à la page 479

La fonction de fournisseur de messagerie IBM MQ dans WLP utilise un composant de IBM MQ appelé adaptateur de ressources IBM MQ . Pour des raisons de compatibilité, les utilisateurs de z/OS Connect V1 doivent utiliser l'adaptateur de ressources IBM WebSphere MQ 7.5 , quelle que soit la version du gestionnaire de files d'attente auquel il se connecte.

V 9.0.1 z/OS Installation de l'adaptateur de ressources IBM MQ

La fonction de fournisseur de messagerie IBM MQ dans WLP utilise un composant de IBM MQ appelé adaptateur de ressources IBM MQ . Pour des raisons de compatibilité, les utilisateurs de z/OS Connect V1 doivent utiliser l'adaptateur de ressources IBM WebSphere MQ 7.5 , quelle que soit la version du gestionnaire de files d'attente auquel il se connecte.

Avant de commencer

Accédez à [Obtention de l'adaptateur de ressources IBM MQ pour WLP](#), qui décrit comment établir un lien vers Fix Central, et téléchargez l'adaptateur de ressources le plus récent.

L'adaptateur de ressources se présente sous la forme suivante: `7.5.0.Fix_level-WS-MQ-Java-InstallRA.jar`. Notez que c'est ce fichier jar auquel la procédure fait référence.

Pourquoi et quand exécuter cette tâche

Cette procédure installe l'adaptateur de ressources IBM MQ dans WLP.

Procédure

1. Extrayez le contenu du fichier jar, accédez au répertoire `wmq` et transférez par FTP le fichier `wmq.jmsra.rar` en binaire dans un répertoire approprié sur le système de fichiers cible.
Ce répertoire est appelé `MQRA_ROOT` pour le reste de ces instructions.
2. Localisez le répertoire Unix System Services Components du gestionnaire de files d'attente cible.
Par exemple, ce répertoire peut être appelé `/mqm/V9R0M1/`. Le répertoire doit contenir un sous-répertoire `java/lib`, qui contient un certain nombre de bibliothèques natives (fichiers `.so`).
3. Editez le fichier `server.xml` du serveur.

Ajoutez les lignes suivantes :

```
<variable name="wmqJmsClient.rar.location"
  value="MQRA_ROOT/wmq.jmsra.rar"/>
<wmqJmsClient nativeLibraryPath="MQJAVA_LIB_DIR"/>
```

où `MQJAVA_LIB_DIR` est le répertoire situé à l'étape «2», à la page 479, par exemple `/usr/lpp/mqm/V9R0M1/java/lib`.

b. Sauvegardez les modifications.

La première ligne indique à WLP où localiser l'adaptateur de ressources IBM MQ .

La deuxième ligne indique à l'adaptateur de ressources IBM MQ où localiser les bibliothèques natives utilisées pour les connexions de liaisons à IBM MQ.

Pour plus d'informations, voir [ClientwmqJmsset Déploiement d'applications JMS dans Liberty pour utiliser le fournisseur de messagerie IBM MQ](#) .

4. Configurez STEPLIB.

Vous devez effectuer cette opération pour que les bibliothèques natives situées à l'étape «2», à la page 479 puissent se connecter au gestionnaire de files d'attente.

Exécutez la commande suivante dans le shell utilisé pour démarrer le serveur WLP:

```
export STEPLIB=HLQ.SCSQAUTH:HLQ.SCSQANLE
```

où *HLQ* est le qualificatif de haut niveau des fichiers contenant l'installation IBM MQ .

Résultats

Vous avez partiellement installé l'adaptateur de ressources IBM MQ . Notez que l'adaptateur de ressources IBM MQ ne sera pas entièrement installé tant que sa fonction parent (wmqJmsClient-1.1) ne sera pas activée.

Que faire ensuite

Installez MQ Service Provider dans WLP.

Tâches associées

«IBM z/OS Connect V1 -Procédure de configuration MQ Service Provider», à la page 477

Présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sur IBM z/OS Connect V1 .

«InstallationWebSphere Application Server Liberty», à la page 477

Configuration requise et procédure d'installation de WebSphere Application Server Liberty (WLP).

«Installation de MQ Service Provider dans WLP», à la page 480

Vous devez installer MQ Service Provider dans WLP avant de l'utiliser.

V 9.0.1 z/OS Installation de MQ Service Provider dans WLP

Vous devez installer MQ Service Provider dans WLP avant de l'utiliser.

Avant de commencer

Procurez-vous et installez le MQ Service Provider à l'aide des informations contenues dans «[Obtention de MQ Service Provider](#)», à la page 466.

Pourquoi et quand exécuter cette tâche

MQ Service Provider est une fonction WLP standard que vous pouvez installer à l'aide de l'une des méthodes décrites dans [Conditionnement et installation de fonctions Liberty](#).

Pour plus d'informations sur la signification de `#{wlp.install.dir}` et de `#{wlp.user.dir}` , voir [Emplacement et propriétés de l'annuaire](#) .

Le texte suivant fournit une vue d'ensemble des méthodes.

Procédure

1. Installez la fonction dans le noyau WLP.



Avertissement : Cette méthode requiert un accès en écriture au répertoire d'installation WLP. Si vous ne disposez pas d'un accès en écriture au répertoire d'installation WLP, l'option «2», à la [page 480](#) est probablement plus appropriée.

Pour ce faire, copiez:

- a) `MQSP_ROOT/v1.0/lib/com.ibm.mq.zosconnect_1.0.0.jar` vers `#{wlp.install.dir}/lib`.
 - b) `MQSP_ROOT/v1.0/lib/features/zosConnectMQ-1.0.mf` vers `#{wlp.install.dir}/lib/features`.
2. Installez la fonction dans la configuration utilisateur.
Pour effectuer cette copie:

- a) `MQSP_ROOT/v1.0/lib/com.ibm.mq.zosconnect_1.0.0.jar` vers `{wlp.user.dir}/extension/lib`.
 - b) `MQSP_ROOT/v1.0/lib/features/zosConnectMQ-1.0.mf` vers `{wlp.user.dir}/extension/lib/features`.
3. Installez la fonction en tant qu'extension de produit.
- Pour ce faire :
- a) Copiez `MQSP_ROOT/mqzosconnect.properties` dans `#{wlp.install.dir}/etc/extensions`.
 - b) Editez le fichier copié.
- Notez que ce fichier est un fichier ASCII. Remplacez `PATH_TO_INSTALL` par `MQSP_ROOT/v1.0` et sauvegardez vos modifications.

Résultats

Vous avez maintenant installé tous les prérequis pour MQ Service Provider.

Remarque : L'approche que vous utilisez pour installer le MQ Service Provider affecte la manière dont il est référencé dans `server.xml`. Des exemples sont présentés dans [«Activation de z/OS Connect et de MQ Service Provider»](#), à la page 481.

Que faire ensuite

Vous devez maintenant activer MQ Service Provider et z/OS Connect.

Tâches associées

[«IBM z/OS Connect V1 -Procédure de configuration MQ Service Provider»](#), à la page 477

Présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sur IBM z/OS Connect V1 .

[«Installation de l'adaptateur de ressources IBM MQ»](#), à la page 479

La fonction de fournisseur de messagerie IBM MQ dans WLP utilise un composant de IBM MQ appelé adaptateur de ressources IBM MQ . Pour des raisons de compatibilité, les utilisateurs de z/OS Connect V1 doivent utiliser l'adaptateur de ressources IBM WebSphere MQ 7.5 , quelle que soit la version du gestionnaire de files d'attente auquel il se connecte.

[«Activation de z/OS Connect et de MQ Service Provider»](#), à la page 481

Actions à effectuer pour activer z/OS Connect et MQ Service Provider.

Activation de z/OS Connect et de MQ Service Provider

Actions à effectuer pour activer z/OS Connect et MQ Service Provider.

Avant de commencer

Vérifiez que vous avez effectué les procédures suivantes:

- [«InstallationWebSphere Application Server Liberty»](#), à la page 477
- [«Installation de l'adaptateur de ressources IBM MQ»](#), à la page 479
- [«Installation de MQ Service Provider dans WLP»](#), à la page 480

Pourquoi et quand exécuter cette tâche

Cette procédure active à la fois z/OS Connect et MQ Service Provider.

Procédure

1. Editez le fichier `server.xml` que vous avez créé et remplacez l'ensemble de l'élément **featureManager** par ces lignes.



Avertissement : Le format exact de la ligne `<feature>zsoConnectMQ-1.0</feature>` dépend de la méthode que vous avez utilisée pour installer MQ Service Provider, décrite dans [«Installation de MQ Service Provider dans WLP»](#), à la page 480.

Si vous avez utilisé:

- Méthode «1», à la page 480, la ligne apparaît sous la forme `<feature>zsoConnectMQ-1.0</feature>`.
- Méthode «2», à la page 480, la ligne apparaît sous la forme `<feature>usr:zsoConnectMQ-1.0</feature>`.
- Méthode «3», à la page 481, la ligne apparaît sous la forme `<feature>mqzsoconnect:zsoConnectMQ-1.0</feature>`.

Modifiez également les entrées suivantes, uniquement si elles n'existent pas déjà.

```
<featureManager>
<feature>zsoConnect-1.0</feature>
<feature>appSecurity-2.0</feature>
<feature>zsoConnectMQ-1.0</feature>
<feature>wmqJmsClient-1.1</feature>
<feature>zsoTransaction-1.0</feature>
</featureManager>
```

2. Configurez la sécurité pour z/OS Connect.

Pour plus d'informations sur la procédure à suivre, voir [Configuration de la sécurité pour z/OS Connect](#).

3. Démarrer le serveur.

Suivez la procédure décrite dans [Démarrage et arrêt d'un serveur à partir de la ligne de commande](#).

Que faire ensuite

Vérifiez que z/OS Connect est correctement configuré.

Tâches associées

«IBM z/OS Connect V1 -Procédure de configuration MQ Service Provider», à la page 477

Présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sur IBM z/OS Connect V1.

«Vérification de la configuration correcte de z/OS Connect», à la page 482

Comment vérifier que z/OS Connect est correctement configuré.

Vérification de la configuration correcte de z/OS Connect

Comment vérifier que z/OS Connect est correctement configuré.

Avant de commencer

Vérifiez que vous avez effectué la procédure décrite dans [«Activation de z/OS Connect et de MQ Service Provider»](#), à la page 481

Pourquoi et quand exécuter cette tâche

z/OS Connect fournit une API RESTful qui peut être utilisée pour interroger les services installés et effectuer des actions d'administration telles que leur arrêt et leur démarrage.

Procédure

1. Emettez une requête HTTP GET sur z/OS Connect pour interroger la liste des services actuellement installés.

Pour ce faire, utilisez un navigateur Web en entrant une URL du formulaire

`https://HOST_NAME:HTTPS_PORT/zosConnect/services`

où `HOST_NAME` et `HTTPS_PORT` sont les valeurs que vous avez entrées à l'étape «2», à la page 478 de «Installation WebSphere Application Server Liberty», à la page 477.

Par exemple :

`https://yourdomainname:12342/zosConnect/services`

2. Entrez l'ID utilisateur et le mot de passe lorsque vous y êtes invité par le navigateur.

Il s'agit des valeurs que vous avez entrées dans l'élément `user` à l'étape «2», à la page 482 de «Activation de z/OS Connect et de MQ Service Provider», à la page 481.

Résultats

Il en résulte la réponse JSON suivante, indiquant que z/OS Connect est en cours d'exécution, mais qu'aucun service n'est installé:

```
-----  
{  
  "zosConnectServices": [  ]  
}
```

Que faire ensuite

«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect V1», à la page 483

Tâches associées

«IBM z/OS Connect V1 -Procédure de configuration MQ Service Provider», à la page 477

Présentation des composants que vous devez installer pour permettre à MQ Service Provider d'être configuré sur IBM z/OS Connect V1 .

Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect V1

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

Avant de commencer

Vérifiez que vous avez configuré tous les composants correctement, comme décrit dans [Activation d' z/OS Connect et de MQ Service Provider](#) et [Vérification de la configuration correcte de z/OS Connect](#)

Procédure

1. Créez une file d'attente appelée `ONE_WAY_QUEUE` sur le gestionnaire de files d'attente z/OS cible, à l'aide de MQSC ou de IBM MQ Explorer.
2. Définissez une fabrique de connexions de fournisseur de messagerie IBM MQ et une file d'attente. Pour ce faire, ajoutez ce qui suit à la fin de `server.xml`, mais à l'intérieur de l'élément `serveur`.

```
-----  
<jmsConnectionFactory id="cf1" jndiName="jms/cf1" connectionManagerRef="ConMgr1">  
  <properties.wmqJms  
    transportType="BINDINGS"  
    queueManager="MQ21"/>  
</jmsConnectionFactory>  
  
<connectionManager id="ConMgr1" maxPoolSize="5"/>
```

```
<jmsQueue id="q1" jndiName="jms/d1">
  <properties.wmqJms
    baseQueueName="ONE_WAY_QUEUE"/>
</jmsQueue>
```

Remarques :

- a. Remplacez la valeur de l'attribut **queueManager** par le nom de gestionnaire de files d'attente cible correct.
 - b. Un **transportType** de liaisons est utilisé. Cela signifie qu'une connexion intermémoire est utilisée pour communiquer avec le gestionnaire de files d'attente. Il s'agit du seul **transportType** pris en charge lors de l'utilisation de MQ Service Provider.
3. Définissez un service MQ Service Provider unidirectionnel simple en ajoutant ce qui suit à `server.xml`, mais à l'intérieur de l'élément `server`.

```
-----
<zosConnectService id="zosconnMQ1"
  invokeURI="/oneWay"
  serviceName="oneWay"
  serviceRef="oneWay"/>
<mqzOSConnectService id="oneWay"
  connectionFactory="jms/cf1"
  destination="jms/d1"/>
-----
```

L'élément `zosConnectService` définit un nouveau service pour z/OS Connect à l'aide du **serviceName** de `oneWay`. Les éléments suivants:

- L'attribut **invokeURI** facilite l'appel du service.
- L'attribut **serviceRef** doit correspondre à l'attribut ID d'un fournisseur de services z/OS Connect, qui, dans ce cas, est fourni par l'élément `mqzOSConnectService`.

L'élément `mqzOSConnectService` définit une instance de service unique fournie par MQ Service Provider.

Les attributs **connectionFactory** et **destination** indiquent à l'instance comment localiser la fabrique de connexions du fournisseur de messagerie IBM MQ et la file d'attente, respectivement.

Pour plus d'informations sur les attributs de cette structure, voir [ElémentmqzOSConnectService](#).

Résultats

Vous avez configuré un service unidirectionnel simple.

Que faire ensuite

Vous devez tester le service.

Tâches associées

«Test du service unidirectionnel sur z/OS Connect V1», à la page 484

Une série d'étapes garantissant que votre service unidirectionnel fonctionne.

  *Test du service unidirectionnel sur z/OS Connect V1*

Une série d'étapes garantissant que votre service unidirectionnel fonctionne.

Avant de commencer

Vérifiez que vous avez correctement effectué la procédure «Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect V1», à la page 483.

Procédure

1. Vérifiez que z/OS Connect reconnaît le nouveau service.

Pour ce faire, réexécutez les étapes décrites dans la rubrique [Vérification de la configuration correcte d' z/OS Connect](#).

Etant donné qu'un service est maintenant défini, vous devriez voir une sortie similaire à la suivante:

```
-----  
{  
  "zosConnectServices": [  
    {  
      "ServiceName": "oneWay",  
      "ServiceDescription": "DATA_UNAVAILABLE",  
      "ServiceProvider": "IBM MQ for z/OS service provider for IBM z/OS Connect" V1.0,  
      "ServiceURL": "https://yourdomainname:12342/zosConnect/services/oneWay"  
    }  
  ]  
}
```

Remarque : **ServiceDescription** est renvoyé en tant que DATA_UNAVAILABLE car l'attribut **serviceDescription** n'a pas été spécifié sur l'élément **zosConnectService**.

2. Emettez une demande HTTP POST au service. Cela entraîne l'envoi d'un nouveau message à la file d'attente ONE_WAY_QUEUE avec le contenu spécifié sur l'autotest à la mise sous tension.

Un bon moyen d'effectuer ce type de test consiste à utiliser un plug-in tel que le [client REST avancé \(ARC\)](#) pour Google Chrome.

- a) Utilisez ARC pour émettre une demande HTTP POST sur l'URL suivante:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

- b) Spécifiez un code JSON valide pour le corps, par exemple:

```
{"name1" : "value1", "name2" : "value2"}
```

Vous devez spécifier deux en-têtes:

- i) Autorisation = Basic *ENCODED_USERID_PASSWORD*

où *ENCODED_USERID_PASSWORD* est un ID utilisateur et un mot de passe codés en base 64, comme décrit dans le [Schéma d'authentification de base](#).

- ii) Type de contenu = application/json

Le premier en-tête est utilisé pour la connexion à z/OS Connect et est une forme codée de l'ID utilisateur et du mot de passe utilisés dans «[Activation de z/OS Connect et de MQ Service Provider](#)», à la page 481.

Le deuxième en-tête informe le serveur que le contenu est JSON, qui est la seule valeur prise en charge lors de l'émission d'appels HTTP POST à MQ Service Provider.

Si vous ne spécifiez pas cet en-tête, une erreur HTTP 415 indiquant un type de support non pris en charge est générée.

L'émission d'une demande HTTP POST vers un service unidirectionnel entraîne le renvoi d'un corps de réponse HTTP vide et d'un code de réponse HTTP 204 (aucun contenu).

ARC génère un message indiquant que la réponse ne contient pas de données.

Pour vérifier qu'un message a été inséré, utilisez un outil tel que l'explorateur IBM MQ pour parcourir le contenu de ONE_WAY_QUEUE. Il doit y avoir un message IBM MQ au format MQSTR dans la file d'attente avec le contenu JSON qui a été envoyé.

3. Utilisez ARC pour émettre une requête HTTP DELETE sur l'URL suivante:

```
https://HOST_NAME:HTTPS_PORT/oneWay
```

Cette opération extrait de façon destructive le message qui vient d'être envoyé au service unidirectionnel.

Cela doit fournir un corps de réponse contenant le contenu JSON qui a été envoyé à l'origine. Vous pouvez utiliser IBM MQ Explorer pour confirmer que le message a été obtenu de façon destructive.

Tâches associées

«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect V1», à la page 483

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

Configuration d'un service IBM MQ bidirectionnelle simple sur z/OS Connect version 1

Comment configurer un service IBM MQ bidirectionnel dans z/OS Connect V1, par rapport à une paire existante de files d'attente IBM MQ appelées MQ_REQUEST_Q et MQ_REPLY_Q.

Avant de commencer

Vérifiez que vous avez configuré tous les composants correctement, comme décrit dans «Activation de z/OS Connect et de MQ Service Provider», à la page 481 et «Vérification de la configuration correcte de z/OS Connect», à la page 482.

Pourquoi et quand exécuter cette tâche

Cette opération est légèrement plus importante que la configuration d'un service unidirectionnel, car vous avez besoin d'une application de back end pour consommer les messages de demande envoyés à l'aide de z/OS Connect et générer des messages de réponse.

Cette tâche suppose que les files d'attente MQ_REQUEST_Q et MQ_REPLY_Q sont utilisées par une application d'arrière-plan existante, par exemple une transaction CICS ou IMS, qui reçoit un message de demande de MQ_REQUEST_Q et génère un message de réponse placé sur MQ_REPLY_Q.

Procédure

1. Définissez une fabrique de connexions de fournisseur de messagerie IBM MQ et deux files d'attente de fournisseur de messagerie IBM MQ sur le serveur z/OS Connect V1.

Les files d'attente du fournisseur de messagerie IBM MQ sont mappées aux files d'attente MQ_REQUEST_Q et MQ_REPLY_Q existantes. Pour ce faire, ajoutez ce qui suit à la fin de `server.xml`, mais à l'intérieur de l'élément `serveur`.

```
<jmsConnectionFactory id="cf2" jndiName="jms/cf2"
  connectionManagerRef="ConMgr2">
  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager id="ConMgr2" maxPoolSize="5"/>

<jmsQueue id="requestQueue" jndiName="jms/requestQueue">
  <properties.wmqJms baseQueueName="MQ_REQUEST_Q"/>
</jmsQueue>

<jmsQueue id="replyQueue" jndiName="jms/replyQueue">
  <properties.wmqJms baseQueueName="MQ_REPLY_Q"/>
</jmsQueue>
```

Remarques :

- a. Remplacez la valeur de l'attribut `queueManager` par le nom de gestionnaire de files d'attente cible correct.

- b. Vous devrez peut-être effectuer une configuration supplémentaire de ces éléments, en fonction de la nature de l'application de back end. Pour plus d'informations, voir [«Configuration de l'MQ Service Provider»](#), à la page 487.
 - c. L'attribut **maxPoolSize** peut avoir besoin d'être ajusté en fonction du nombre de demandes simultanées. Pour plus d'informations sur cet attribut, voir [Configuration du regroupement de connexions pour les connexions JMS](#).
2. Définissez le service IBM MQ bidirectionnel en ajoutant ce qui suit à `server.xml`, mais à l'intérieur de l'élément `server`.

```
-----  
<zosConnectService id="zosconnMQ2"  
    invokeURI="/twoWay"  
    serviceName="twoWay"  
    serviceRef="twoWay"/>  
  
<mqzOSConnectService id="twoWay"  
    connectionFactory="jms/cf2"  
    destination="jms/requestQueue"  
    replyDestination="jms/replyQueue"  
    waitInterval="10000"/>  
-----
```

Remarques :

- a. Vous devrez peut-être effectuer une configuration supplémentaire de ces éléments, en fonction de la nature de l'application de back end. Par exemple, vous pouvez avoir besoin d'une transformation de données. Pour plus d'informations, voir [«Configuration de l'MQ Service Provider»](#), à la page 487.
- b. Vous devrez peut-être ajuster la valeur de **waitInterval**. Pour plus d'informations, voir [waitInterval](#).
- c. Dans cet exemple, on suppose que les messages placés sur la destination de réponse, par l'application de back end, ont un ID de corrélation identique à l'ID de message sur la destination de demande.

Si ce n'est pas le cas, vous devez ajouter l'attribut **replySelection** à l'élément `mqzOSConnectService` et définir la valeur de manière appropriée.

Pour plus d'informations, voir [replySelection](#).

Résultats

Vous avez configuré un service bidirectionnel dans z/OS Connect V1, qui peut être utilisé avec une application de back end existante qui utilise une paire de files d'attente IBM MQ.

Que faire ensuite

Vous pouvez appeler le service bidirectionnel en émettant une commande HTTP POST sur l'URL `https://host_name:port_no/twoWay`.

Les étapes à suivre sont similaires à celles décrites dans [«Test du service unidirectionnel sur z/OS Connect V1»](#), à la page 484, sauf que seule l'instruction HTTP POST est prise en charge et que vous devez envoyer des données de contenu JSON appropriées à l'application de back end existante.

Tâches associées

[«Configuration d'un service MQ Service Provider unidirectionnelle simple sur z/OS Connect V1»](#), à la page 483

Pour configurer un service MQ Service Provider unidirectionnel simple, procédez comme suit:

Configuration de l'MQ Service Provider

Vous devez prendre en compte plusieurs domaines avant d'utiliser le MQ Service Provider.

Cette section contient :

- «Remarques relatives à la sécurité lors de l'utilisation de MQ Service Provider», à la page 488
- «Utilisation des transformations de données avec MQ Service Provider», à la page 491
- «Utilisation des applications MQI», à la page 496

V 9.0.1

z/OS

Remarques relatives à la sécurité lors de l'utilisation de MQ

Service Provider

Il existe deux ensembles de considérations de sécurité pour le MQ Service Provider.

- Indique si un utilisateur donné peut soumettre des demandes RESTful aux services MQ Service Provider exposés par z/OS Connect.
- Indique si un service MQ Service Provider donné est autorisé à accéder au gestionnaire de files d'attente et à ses ressources.

Indique si un utilisateur donné peut soumettre des demandes RESTful aux services MQ Service Provider exposés à partir de z/OS Connect

Cela est contrôlé par la configuration dans le fichier `server.xml` de z/OS Connect.

Pour z/OS Connect EE, voir [Configuration de la sécurité pour z/OS Connect EE](#).

Pour z/OS Connect V1, voir [Configuration de la sécurité pour z/OS Connect](#).

Indique si un service MQ Service Provider donné est autorisé à accéder au gestionnaire de files d'attente et à ses ressources

MQ Service Provider est fondamentalement une application JMS, basée sur le fournisseur de messagerie IBM MQ fourni avec WLP, qui se connecte à un ou plusieurs gestionnaires de files d'attente à l'aide de connexions en mode liaison.

Par conséquent, MQ Service Provider peut être sécurisé de la même manière que les applications qui partagent ces caractéristiques. Cette rubrique décrit ces similitudes et souligne les différences.

MQ Service Provider étant une application de liaisons, il se connecte à un gestionnaire de files d'attente et présente un ID utilisateur et un mot de passe facultatif. Elles sont éventuellement validées à l'aide de l'authentification de connexion.

S'il est valide, l'utilisateur est autorisé à se connecter en fonction de la configuration de sécurité du gestionnaire de files d'attente. Pour plus d'informations, voir [Configuration de la sécurité sous z/OS](#).

Les attributs de configuration de l'élément de service `mqzOSConnect` et du sous-élément `properties.wmqJMS` de l'élément `jmsConnectFactory` dans `server.xml` ont une incidence sur l'ID utilisateur et le mot de passe facultatif présentés au gestionnaire de files d'attente.

Les différentes possibilités sont détaillées dans le tableau suivant.



Avertissement : Il existe un ordre de priorité parmi les propriétés de configuration de la sécurité. Les attributs `mqzOSConnectService` remplacent les attributs `properties.wmqJms` et l'attribut `useCallerPrincipal` remplace tous les autres attributs.

Les deux attributs de mot de passe peuvent être fournis en texte en clair ou au format codé. Vous devez utiliser le format codé, car toute personne ayant accès à `server.xml` peut afficher le mot de passe en texte en clair.

Les deux versions de z/OS Connect sont fournies avec un outil appelé `securityUtility` qui peut être utilisé pour coder les mots de passe. Pour plus d'informations, voir [Liberty: Commande securityUtility](#).

Tableau 28. Autorisations de service

| Élément de service mqzOSConnect | | Élément propriétés.wmqJms | | Résultat |
|---------------------------------|-----------------------------------|---------------------------|-------------------|---|
| useCallerPrincipal | Nom d'utilisateur et mot de passe | userName | password | |
| non défini / faux | non défini / vide | non défini / vide | non défini / vide | Le nom d'utilisateur associé à l'espace adresse z/OS Connect est présenté au gestionnaire de files d'attente à des fins d'autorisation et d'authentification. Aucun mot de passe n'est présenté. |
| non défini / faux | non défini / vide | set | non défini / vide | Le nom d'utilisateur sur l'élément propriétés.wmqJms est présenté au gestionnaire de files d'attente à des fins d'autorisation et d'authentification. Aucun mot de passe n'est présenté. |
| non défini / faux | non défini / vide | set | set | Le nom d'utilisateur et le mot de passe de l'élément propriétés.wmqJms sont présentés au gestionnaire de files d'attente à des fins d'autorisation et d'authentification. |
| non défini / faux | ensemble des deux valeurs | ignoré si défini | ignoré si défini | Le nom d'utilisateur et le mot de passe de l'élément mqzOSConnectService sont présentés au gestionnaire de files d'attente à des fins d'autorisation et d'authentification. |
| conforme | ignoré si défini | ignoré si défini | ignoré si défini | Le principal d'utilisateur qui a été authentifié auprès de z/OS Connect est utilisé et son nom d'utilisateur est présenté au gestionnaire de files d'attente à des fins d'autorisation et d'authentification. Aucun mot de passe n'est présenté. Des informations sur la configuration de la sécurité pour z/OS Connect EE sont disponibles à l'adresse ¹ . Pour plus d'informations sur z/OS Connect version 1, voir ² . |

Remarques :

1. [Configuration de la sécurité pour z/OS Connect EE](#)
2. [Configuration de la sécurité pour z/OS Connect](#)

Exemples

Il s'agit d'un service unidirectionnel dans lequel la fabrique de connexions et le mqzOSConnectService spécifient tous deux un **userID** et un **password**. Dans les deux cas, les mots de passe sont codés. Les éléments **userID** et **password** définis dans la définition mqzOSConnectService sont utilisés.

z/OS Connect V1

```
<zosConnectService
id="samplezOSConnectService1"
invokeURI="/samplezOSConnectService1"
serviceName="samplezOSConnectService1_name"
serviceRef="samplezOSConnectService1_MQ"/>

<mqzOSConnectService
id="samplezOSConnectService1_MQ"
connectionFactory="jms/sampleCF1"
userName="bill"
password="{aes}AJ+DdZ+1u0KEG5KIwUz4LvHBAQ8nTd3y8K8HAI+48Tt"
destination="jms/sampleQ1"/>

<jmsConnectionFactory
id="sampleCF1"
jndiName="jms/sampleCF1"
connectionManagerRef="sampleCF2ConnectionManager1">

<properties.wmqJms
transportType="BINDINGS"
queueManager="MQ21"
userName="matt"
password="{xor}GBMeEg9uERg="/>
</jmsConnectionFactory>

<jmsQueue
id="sampleQ1"
jndiName="jms/sampleQ1">

<properties.wmqJms
baseQueueName="SampleQ1"/>
</jmsQueue>
```

z/OS Connect EE

```
<zosconnect_zosConnectService
id="samplezOSConnectService1"
invokeURI="/samplezOSConnectService1"
serviceName="samplezOSConnectService1_name"
serviceRef="samplezOSConnectService1_MQ"/>

<mqzosconnect_mqzOSConnectService
id="samplezOSConnectService1_MQ"
connectionFactory="jms/sampleCF1"
userName="bill"
password="{aes}AJ+DdZ+1u0KEG5KIwUz4LvHBAQ8nTd3y8K8HAI+48Tt"
destination="jms/sampleQ1"/>

<jmsConnectionFactory
id="sampleCF1"
jndiName="jms/sampleCF1"
connectionManagerRef="sampleCF2ConnectionManager1">

<properties.wmqJms
transportType="BINDINGS"
queueManager="MQ21"
userName="matt"
password="{xor}GBMeEg9uERg="/>
</jmsConnectionFactory>

<jmsQueue
id="sampleQ1"
jndiName="jms/sampleQ1">

<properties.wmqJms
baseQueueName="SampleQ1"/>
</jmsQueue>
```

Cet exemple illustre une définition de service unidirectionnelle qui utilise les mêmes objets JMS (la file d'attente `sampleQ1` et la fabrique de connexions `sampleCF1`) que le premier exemple. Le principal authentifié auprès de z/OS Connect est présenté au gestionnaire de files d'attente car **`useCallerPrincipal="true"`** est spécifié.

z/OS Connect V1

```
<zosConnectService
id="samplezOSConnectService2"
invokeURI="/samplezOSConnectService2"
serviceName="samplezOSConnectService2_name"
serviceRef="samplezOSConnectService2_MQ"/>

<mqzOSConnectService
id="samplezOSConnectService2_MQ"
connectionFactory="jms/sampleCF1"
destination="jms/sampleQ1"
useCallerPrincipal="true"/>
```

z/OS Connect EE

```
<zosconnect_zosConnectService
id="samplezOSConnectService2"
invokeURI="/samplezOSConnectService2"
serviceName="samplezOSConnectService2_name"
serviceRef="samplezOSConnectService2_MQ"/>

<mqzosconnect_mqzOSConnectService
id="samplezOSConnectService2_MQ"
connectionFactory="jms/sampleCF1"
destination="jms/sampleQ1"
useCallerPrincipal="true"/>
```

V 9.0.1

z/OS

Utilisation des transformations de données avec MQ Service

Provider

z/OS Connect permet de transformer des données JSON dans un format arbitraire avant d'appeler un actif z/OS de back end, tel qu'une transaction CICS , et de transformer la réponse de l'actif z/OS en JSON.

Cette capacité est fournie par des fournisseurs de transformation de données connectables. Un fournisseur intégré permet à JSON d'être transformé vers ou depuis des structures COBOL, PLI ou C.

MQ Service Provider fonctionne avec les transformations de données z/OS Connect, mais vous devez prendre en compte un certain nombre de considérations.

Plus d'informations sur les transformations de données z/OS Connect

Pour z/OS Connect Enterprise Edition, voir [Définition des transformateurs de données](#).

Pour z/OS Connect version 1, voir [Définition des transformations de contenu de message z/OS Connect](#).

envoi de messages IBM MQ

Lorsqu'un service MQ Service Provider est configuré avec une transformation de données et qu'il va envoyer un message à un gestionnaire de files d'attente (par exemple, s'il s'agit d'un service unidirectionnel ou bidirectionnel ayant reçu une demande HTTP POST contenant JSON), MQ Service Provider effectue les étapes suivantes:

1. Extrait le contenu JSON de la demande HTTP.
2. Transmet le contenu à z/OS Connect pour effectuer la transformation de données de JSON en tableau d'octets.
3. Extrait le résultat de la transformation de données et l'envoie au gestionnaire de files d'attente sous la forme d'un message JMS BytesMessage.

Par défaut, le message envoyé comporte une zone MQMD **Format** vide. Dans la plupart des cas, cela n'est pas approprié. Vous pouvez donc définir l'attribut `mqmdFormat` dans l' [élément de servicemqzOSConnect](#) sur une valeur appropriée.

La prise en charge de la transformation de données intégrée génère toujours une sortie avec un CCSID de 37. Ces informations doivent être spécifiées dans la zone MQMD `CodedCharSetId` , faute

de quoi l'application provenant de la file d'attente risque de ne pas pouvoir décoder le message. Pour ce faire, définissez l'attribut CCSID sur la file d'attente référencée par l'élément de service mqzOSConnectService .

L'exemple de configuration suivant illustre la configuration appropriée pour un service unidirectionnel, qui va être utilisé pour envoyer des messages à une file d'attente appelée SampleQ1.

Les messages doivent être envoyés avec une zone MQMD Format de *AFORMAT* et une zone CodedCharSetId de 37.

L'élément **zosConnectDataXform** indique à z/OS Connect où localiser la configuration pour les transformations de données et est référencé à l'aide de l'attribut **dataXformRef** de l'élément **zosConnectService** .

Pour la configuration d'un service bidirectionnelle, voir [«Réception de messages depuis IBM MQ»](#), à la page 492.

```
<jmsConnectionFactory
  id="sampleCF1"
  jndiName="jms/sampleCF1"
  connectionManagerRef="sampleCF1ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF1ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ1"
  jndiName="jms/sampleQ1">

  <properties.wmqJms
    baseQueueName="SampleQ1"
    CCSID="37" />
</jmsQueue>

<zosConnectService
  id="samplezOSConnectService1"
  invokeURI="/samplezOSConnectService1"
  serviceName="samplezOSConnectService1_name"
  serviceRef="samplezOSConnectService1_MQ"
  dataXformRef="xformJSON2Byte"/>

<mqzOSConnectService
  id="samplezOSConnectService1_MQ"
  connectionFactory="jms/sampleCF1"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ1" />

<zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json" />
```

Réception de messages depuis IBM MQ

Lorsqu'une instance MQ Service Provider est configurée avec une transformation de données et qu'elle va recevoir un message d'un gestionnaire de files d'attente (par exemple, s'il s'agit d'un service unidirectionnel exécutant une opération HTTP GET ou DELETE, ou d'un service bidirectionnel ayant reçu une demande HTTP POST), MQ Service Provider effectue les étapes suivantes.



Avertissement : Si vous utilisez un service bidirectionnelle, le service aura déjà effectué les étapes décrites dans [«envoi de messages IBM MQ»](#), à la page 491.

1. Extrait le message de la file d'attente.

2. Vérifie que le message est un JMS **BytesMessage** ou un JMS **TextMessage**. Si le message n'est ni l'un ni l'autre, une erreur est générée et renvoyée à l'appelant. "
3. Transmet la charge de message à z/OS Connect pour effectuer une transformation de données à partir d'un tableau d'octets vers JSON.
4. Prend le résultat de la transformation de données et le renvoie en tant que réponse de la méthode HTTP.

En fonction du type de message reçu, vous devrez peut-être effectuer une configuration supplémentaire. En effet, MQ Service Provider doit convertir la charge de message reçue dans le format correct pour la transmission à la transformation de données configurée.

La transformation de données par défaut s'attend à ce que le contenu soit dans le CCSID 37, mais il se peut que l'actif z/OS ne génère pas de messages dans ce CCSID.

La configuration requise varie selon qu'un **BytesMessage** ou un **TextMessage** est reçu et que vous utilisez z/OS Connect V1 ou z/OS Connect EE.

BytesMessage reçu sur z/OS Connect V1

Si un **BytesMessage** doit être reçu, vous pouvez spécifier les attributs **receiveConversion**= "QMGR" et **receiveCCSID**= "37" dans la définition de file d'attente utilisée pour recevoir le message.

Cela est représenté dans l'exemple suivant. Dans ce cas, les attributs **receiveConversion** et **receiveCCSID** sont définis pour la définition `sampleQ2Receive`.

```
<jmsConnectionFactory
  id="sampleCF2"
  jndiName="jms/sampleCF2"
  connectionManagerRef="sampleCF2ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF2ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ2Send"
  jndiName="jms/sampleQ2Send">

  <properties.wmqJms
    baseQueueName="SampleQ2Send"
    CCSID="37" />
</jmsQueue>

<jmsQueue id="sampleQ2Receive"
  jndiName="jms/sampleQ2Receive">

  <properties.wmqJms
    baseQueueName="SampleQ2Receive"
    receiveCCSID="37"
    receiveConversion="QMGR" />
</jmsQueue>

<zosConnectService
  id="samplezOSConnectService2"
  invokeURI="/samplezOSConnectService2"
  serviceName="samplezOSConnectService2_name"
  serviceRef="samplezOSConnectService2_MQ"
  dataXformRef="xformJSON2Byte"/>

<mqzOSConnectService
  id="samplezOSConnectService2_MQ"
  connectionFactory="jms/sampleCF2"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ2Send"
  replyDestination="jms/sampleQ3Receive"/>

<zosConnectDataXform id="xformJSON2Byte"
```

```

bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
responseSchemaLoc="/XFORM_ROOT/json"
responseSchemaSuffix=".json"/>

```

BytesMessage reçu sur z/OS Connect EE

Si un **BytesMessage** doit être reçu, vous pouvez spécifier les attributs **receiveConversion**= "QMGR" et **receiveCCSID**= "37" dans la définition de file d'attente utilisée pour recevoir le message.

Cela est représenté dans l'exemple suivant. Dans ce cas, les attributs **receiveConversion** et **receiveCCSID** sont définis pour la définition `sampleQ2Receive`.

```

<jmsConnectionFactory
  id="sampleCF2"
  jndiName="jms/sampleCF2"
  connectionManagerRef="sampleCF2ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21"/>
</jmsConnectionFactory>

<connectionManager
  id="sampleCF2ConnectionManager"
  maxPoolSize="5"/>

<jmsQueue id="sampleQ2Send"
  jndiName="jms/sampleQ2Send">

  <properties.wmqJms
    baseQueueName="SampleQ2Send"
    CCSID="37"/>
</jmsQueue>

<jmsQueue id="sampleQ2Receive"
  jndiName="jms/sampleQ2Receive">

  <properties.wmqJms
    baseQueueName="SampleQ2Receive"
    receiveCCSID="37"
    receiveConversion="QMGR"/>
</jmsQueue>

<zosconnect_zosConnectService
  id="samplezOSConnectService2"
  invokeURI="/samplezOSConnectService2"
  serviceName="samplezOSConnectService2_name"
  serviceRef="samplezOSConnectService2_MQ"
  dataXformRef="xformJSON2Byte"/>

<mqzosconnect_mqzOSConnectService
  id="samplezOSConnectService2_MQ"
  connectionFactory="jms/sampleCF2"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ2Send"
  replyDestination="jms/sampleQ3Receive"/>

<zosconnect_zosConnectDataXform id="xformJSON2Byte"
  bindFileLoc="/XFORM_ROOT/bindfiles" bindFileSuffix=".bnd"
  requestSchemaLoc="/XFORM_ROOT/json" requestSchemaSuffix=".json"
  responseSchemaLoc="/XFORM_ROOT/json"
  responseSchemaSuffix=".json"/>

```

TextMessage reçu sur z/OS Connect V1

Si un **TextMessage** doit être reçu, vous devez convertir le message au CCSID attendu par la transformation de données, par défaut 37.

Si vous utilisez une transformation de données personnalisée et que la transformation attend un CCSID différent de 37, vous pouvez spécifier l'attribut `receiveTextCCSID` dans l'élément de `servicemqzOSConnect` pour définir un CCSID approprié.

Cela est représenté dans l'exemple suivant. Dans ce cas, l'exemple de définition `zOSConnectService3_MQ` comporte l'attribut `receiveTextCCSID` dans l'attribut `mqzOSConnectService` élément défini sur 1208 (UTF-8).

```
<jmsConnectionFactory
  id="sampleCF3"
  jndiName="jms/sampleCF3"
  connectionManagerRef="sampleCF3ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF3ConnectionManager"
  maxPoolSize="5" />

<jmsQueue id="sampleQ3Send"
  jndiName="jms/sampleQ3Send">

  <properties.wmqJms
    baseQueueName="SampleQ3Send"
    CCSID="37" />
</jmsQueue>

<jmsQueue id="sampleQ3Receive"
  jndiName="jms/sampleQ3Receive">

  <properties.wmqJms
    baseQueueName="SampleQ3Receive" />
</jmsQueue>

<zosConnectService
  id="samplezOSConnectService3"
  invokeURI="/samplezOSConnectService3"
  serviceName="samplezOSConnectService3_name"
  serviceRef="samplezOSConnectService3_MQ"
  dataXformRef="customDataXForm" />

<mqzOSConnectService
  id="samplezOSConnectService3_MQ"
  connectionFactory="jms/sampleCF3"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ3Send"
  replyDestination="jms/sampleQ3Receive"
  receiveTextCCSID="1208" />
```

TextMessage reçu sur z/OS Connect EE

Si un **TextMessage** doit être reçu, vous devez convertir le message au CCSID attendu par la transformation de données, par défaut 37.

Si vous utilisez une transformation de données personnalisée et que la transformation attend un CCSID différent de 37, vous pouvez spécifier l'attribut `receiveTextCCSID` dans l'élément de `servicemqzOSConnect` pour définir un CCSID approprié.

Cela est représenté dans l'exemple suivant. Dans ce cas, l'exemple de définition `zOSConnectService3_MQ` comporte l'attribut `receiveTextCCSID` dans l'attribut `mqzOSConnectService` élément défini sur 1208 (UTF-8).

```
<jmsConnectionFactory
  id="sampleCF3"
  jndiName="jms/sampleCF3"
  connectionManagerRef="sampleCF3ConnectionManager">

  <properties.wmqJms
    transportType="BINDINGS"
    queueManager="MQ21" />
</jmsConnectionFactory>

<connectionManager
  id="sampleCF3ConnectionManager"
  maxPoolSize="5" />
```

```

<jmsQueue id="sampleQ3Send"
  jndiName="jms/sampleQ3Send">
  <properties.wmqJms
    baseQueueName="SampleQ3Send"
    CCSID="37"/>
</jmsQueue>

<jmsQueue id="sampleQ3Receive"
  jndiName="jms/sampleQ3Receive">
  <properties.wmqJms
    baseQueueName="SampleQ3Receive"/>
</jmsQueue>

<zosconnect_zosConnectService
  id="samplezOSConnectService3"
  invokeURI="/samplezOSConnectService3"
  serviceName="samplezOSConnectService3_name"
  serviceRef="samplezOSConnectService3_MQ"
  dataXformRef="customDataXForm"/>

<mqzosconnect_mqzOSConnectService
  id="samplezOSConnectService3_MQ"
  connectionFactory="jms/sampleCF3"
  mqmdFormat="AFORMAT"
  destination="jms/sampleQ3Send"
  replyDestination="jms/sampleQ3Receive"
  receiveTextCCSID="1208"/>

```

V 9.0.1

z/OS

Utilisation des applications MQI

MQ Service Provider fournit une interface RESTful aux applications existantes qui utilisent IBM MQ. Ces applications peuvent interagir avec IBM MQ à l'aide de IBM MQ classes for JMS ou de l'interface MQI (Message Queue Interface)

Par défaut, IBM MQ classes for JMS envoie des messages qui incluent un en-tête MQRFH2. Toutefois, la plupart des applications MQI n'utilisent pas d'en-têtes MQRFH2.

Si une application IBM MQ classes for JMS interagit avec une application MQI qui n'a pas été conçue pour fonctionner avec des en-têtes MQRFH2, une configuration est requise pour empêcher l'IBM MQ classes for JMS envoi de messages contenant un en-tête MQRFH2. Pour plus d'informations, voir [Mappage de messages JMS vers des messages IBM MQ](#).

Si MQ Service Provider envoie des messages aux files d'attente, qui seront utilisées par les applications MQI, qui n'attendent pas d'en-tête MQRFH2, vous devez configurer z/OS Connect pour empêcher l'envoi d'un en-tête MQRFH2.

Pour obtenir cette configuration, ajoutez un attribut **targetClient= "MQ"** à la file d'attente du fournisseur de messagerie IBM MQ appropriée dans `server.xml`.

L'exemple de configuration suivant illustre un service MQ Service Provider unidirectionnel avec un **id** de `mqiService`. A son tour, il a été configuré pour utiliser la file d'attente du fournisseur de messagerie MQ avec un **id** de `mqiQueue`.

`mqiQueue` a été configuré avec l'attribut **targetClient= "MQ"**. Cela signifie que lorsqu'une application envoie un message JMS à cette file d'attente (par exemple, si une application envoie une demande HTTP POST à `mqiService`), aucun en-tête MQRFH2 n'est ajouté au message.

z/OS Connect V1

```

<mqzOSConnectService
  id="mqiService"
  connectionFactory="jms/mqiCF"
  destination="jms/mqiQueue"/>

<jmsQueue
  id="mqiQueue"
  jndiName="jms/mqiQueue">

<properties.wmqJms

```



```
    baseQueueName="MQIQueue
    targetClient = "MQ"/>
</jmsQueue>
```

z/OS Connect EE

```
<mqzosconnect_mqzOSConnectService
  id="mqiService"
  connectionFactory="jms/mqiCF"
  destination="jms/mqiQueue"/>

<jmsQueue
  id="mqiQueue"
  jndiName="jms/mqiQueue">

  <properties.wmqJms
    baseQueueName="MQIQueue
    targetClient = "MQ"/>
</jmsQueue>
```


Remarques

:NONE.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Pour plus de détails, référez-vous aux documents d'annonce disponibles dans votre pays, ou adressez-vous à votre partenaire commercial IBM. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout autre élément fonctionnellement équivalent peut être utilisé, s'il n'enfreint aucun droit d'IBM. 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 EMEA Director of Licensing
IBM Corporation
Tour Descartes
Armonk, NY 10504-1785
U.S.A.

Pour toute demande d'informations relatives au jeu de caractères codé sur deux octets, contactez le service de propriété intellectuelle IBM ou envoyez vos questions par courrier à l'adresse suivante :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japon

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 logiciels décrits dans ce document.

Les références à des sites Web non IBM sont fournies à titre d'information uniquement et n'impliquent en aucun cas une adhésion aux données qu'ils contiennent. 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 de WebSphere 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 d'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 (<http://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: