

9.4

Configuration 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 1113.

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

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

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

Table des matières

Configuration.....	7
Création de gestionnaires de files d'attente sur Multiplatforms.....	7
Répertoire éphémère configurable.....	11
Répertoire userdata.....	12
Création d'un gestionnaire de files d'attente par défaut.....	12
Faire d'un gestionnaire de files d'attente existant la valeur par défaut.....	14
Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente.....	15
Déplacement d'un gestionnaire de files d'attente sur un système d'exploitation différent.....	15
Configuration des connexions entre le client et le serveur.....	17
Type de communication à utiliser.....	17
Comment configurer un IBM MQ MQI client.....	20
Configuration d'un client transactionnel étendu.....	21
Définition de canaux MQI.....	32
Création et utilisation de canaux AMQP.....	32
Création de définitions de connexion serveur et de connexion client sur différentes plateformes..	38
Création de définitions de connexion serveur et de connexion client sur le serveur.....	44
Programmes d'exit de canal pour les canaux MQI.....	62
Connecting a client to a queue sharing group.....	66
Utilisation des variables d'environnement IBM MQ.....	67
Description des variables d'environnement.....	68
Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms.....	96
Fichier de configuration IBM MQ , mqs.ini.....	98
Fichiers de configuration du gestionnaire de files d'attente, qm.ini.....	111
Fichier de configuration de l'installation, mqinst.ini.....	174
Fichier de configuration IBM MQ MQI client , mqclient.ini.....	174
Fichier de configuration de trace d'activité, mqat.ini.....	209
Configuration de la file d'attente répartie.....	212
Techniques de mise en file d'attente répartie IBM MQ.....	213
Introduction à la gestion de files d'attente réparties.....	235
Surveillance et contrôle des canaux sur AIX, Linux, and Windows.....	268
Surveillance et contrôle des canaux sur IBM i.....	293
Configuration d'un cluster de gestionnaires de files d'attente.....	316
Configuration d'un cluster uniforme.....	445
Configuration de la messagerie de type publication/abonnement.....	468
Définition des attributs de messages de publication/d'abonnement en file d'attente.....	468
Démarrage de la publication/l'abonnement en file d'attente.....	470
Arrêt de la publication/l'abonnement en file d'attente.....	470
Ajout d'un flux.....	471
Suppression d'un flux.....	472
Ajout d'un point d'abonnement.....	472
Configuration des réseaux de publication/abonnement réparti.....	473
Configuration de plusieurs installations.....	492
Connexion d'applications dans un environnement avec plusieurs installations.....	492
Changement d'installation principale.....	500
Association d'un gestionnaire de files d'attente à une installation.....	502
Recherche d'installations d'IBM MQ sur un système.....	503
Configuration de la haute disponibilité, de la reprise et du redémarrage.....	504
reconnexion client automatique.....	505
Console message monitoring.....	512
Configurations à haute disponibilité.....	516
Consignation: S'assurer que les messages ne sont pas perdus.....	703
Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ.....	737

Modifications apportées à la reprise sur incident de cluster sur les serveurs sur Multiplatforms..	745
Configuration des ressources JMS et Jakarta Messaging.....	747
Configuration de fabriques de connexions et de destinations dans un espace de nom JNDI.....	748
Configuration d'objets JMS 2.0 à l'aide de IBM MQ Explorer.....	752
Configuration des objets JMS et Jakarta Messaging à l'aide des outils d'administration.....	753
Configuration des ressources JMS 2.0 dans WebSphere Application Server.....	764
Configuration de WebSphere Application Server pour utiliser le dernier niveau de maintenance de l'adaptateur de ressources.....	774
Configuration de la propriété JMS PROVIDERVERSION	776
Suppression d'abonnements durables WebSphere Application Server.....	784
Configuration de Managed File Transfer.....	787
Options de configuration de MFT sur Multiplatforms.....	787
MFT configuration options on z/OS.....	789
Téléchargement et configuration de Redistributable Managed File Transfer components.....	790
Creating an MFT Agent or Logger command data set.....	795
Configuring Managed File Transfer for z/OS.....	796
Configuration de MFT sur IBM i.....	828
Configuration de MFT pour la première utilisation.....	830
Configuration des gestionnaires de files d'attente d'agent MFT.....	841
Configuration d'un consignateur MFT.....	852
Configuration du pont Connect:Direct.....	877
Configuration de IBM MQ Console et de REST API.....	883
Configuration de base pour le serveur mqweb.....	883
Configuration du IBM MQ Web Server autonome.....	887
Configuration de la sécurité.....	889
Configuration du nom d'hôte HTTP.....	889
Configuration des ports HTTP et HTTPS.....	890
Configuration du délai d'attente de réponse.....	891
Configuration du démarrage automatique.....	892
Configuration de la journalisation.....	893
Configuration du jeton LTPA.....	897
Configuration du comportement de connexion du gestionnaire de files d'attente éloignées pour IBM MQ Console.....	899
Configuration de la passerelle administrative REST API.....	901
Configuration de messaging REST API.....	902
Configuration de REST API pour MFT.....	909
Optimisation de la machine virtuelle Java du serveur mqweb.....	914
Structure de fichier du composant d'installation IBM MQ Console et REST API.....	916
Sauvegarde et restauration de la configuration de votre serveur mqweb.....	918
Définition d'une connexion Aspera gateway sur les plateformes Linux ou Windows.....	920
Configuration de IBM MQ pour une utilisation avec le service de décompte IBM Cloud Private.....	925
Configuration d'un gestionnaire de files d'attente à utiliser avec l'instance de service de décompte sous IBM Cloud Private.....	926
Connexion au service de décompte IBM Cloud Private via un proxy HTTP.....	929
Traitement des incidents liés à la connexion au service de décompte.....	930
Configuring queue managers on z/OS.....	930
Preparing to customize queue managers on z/OS.....	931
Setting up IBM MQ for z/OS.....	935
Testing a queue manager on z/OS.....	1001
Setting up communications with other queue managers on z/OS.....	1009
Using IBM MQ with IMS.....	1039
Using IBM MQ with CICS.....	1047
Upgrading and applying service to Language Environment or z/OS Callable Services.....	1047
Using OTMA exits in IMS.....	1049
Using IBM z/OSMF to automate IBM MQ	1053
Activation des agents MFT pour la connexion à des gestionnaires de files d'attente z/OS éloignées.....	1064
Configuration de IBM MQ Internet Pass-Thru.....	1065

Prise en charge HTTP dans MQIPT.....	1065
Prise en charge de SOCKS dans MQIPT.....	1067
Prise en charge de SSL/TLS dans MQIPT.....	1068
Java security manager dans MQIPT.....	1098
Exits de sécurité dans MQIPT.....	1100
Contrôle de numéro de port dans MQIPT.....	1104
Chiffrement des mots de passe stockés dans MQIPT.....	1105
Autres considérations de sécurité pour MQIPT.....	1106
Journaux de connexion dans MQIPT.....	1107
Configuration de IBM MQ Internet Pass-Thru à l'aide de conteneurs.....	1109
Configuration des files d'attente de flux.....	1109

Remarques..... 1113

Documentation sur l'interface de programmation.....	1114
Marques.....	1114

Configuration de IBM MQ

Créez un ou plusieurs gestionnaires de files d'attente sur un ou plusieurs ordinateurs et configurez-les sur vos systèmes de développement, de test et de production pour traiter les messages qui contiennent vos données métier.

Pourquoi et quand exécuter cette tâche

Avant de configurer IBM MQ, lisez les informations relatives aux concepts IBM MQ dans la rubrique [IBM MQ Présentation technique](#). Pour plus d'informations sur la planification de votre environnement IBM MQ, voir [Planification](#).

Il existe un certain nombre de méthodes différentes que vous pouvez utiliser pour créer, configurer et administrer vos gestionnaires de files d'attente et leurs ressources associées dans IBM MQ. Ces méthodes incluent des interfaces de ligne de commande, une interface graphique et une API d'administration. Pour plus d'informations sur ces interfaces, voir [Administration d' IBM MQ](#).

Pour savoir comment créer, démarrer, arrêter et supprimer un gestionnaire de files d'attente, voir [«Création de gestionnaires de files d'attente sur Multiplatforms»](#), à la page 7.

Pour plus d'informations sur la création des composants requis pour connecter vos installations et applications IBM MQ ensemble, voir [«Configuration de la file d'attente répartie»](#), à la page 212.

Pour savoir comment connecter vos clients à un serveur IBM MQ à l'aide de différentes méthodes, voir [«Configuration des connexions entre le client et le serveur»](#), à la page 17.

Pour savoir comment configurer un cluster de gestionnaires de files d'attente, voir [«Configuration d'un cluster de gestionnaires de files d'attente»](#), à la page 316.

Vous pouvez modifier le comportement d' IBM MQ ou d'un gestionnaire de files d'attente en modifiant les informations de configuration. Pour plus d'informations, voir [«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms»](#), à la page 96. En règle générale, il n'est pas nécessaire de redémarrer un gestionnaire de files d'attente pour que les modifications de configuration soient prises en compte, sauf lorsqu'elles sont indiquées dans la documentation du produit.

 Pour savoir comment configurer IBM MQ for z/OS, voir [«Configuring queue managers on z/OS»](#), à la page 930.

Concepts associés

[Présentation technique du IBM MQ](#)

Tâches associées

[Administration des objets IBM MQ locaux](#)

[Administration des objets IBM MQ éloignés](#)

 [Administration de IBM i](#)

 [Administration de IBM MQ for z/OS](#)

[Planification](#)

 [Planification de votre environnement IBM MQ sur z/OS](#)

[«Configuring queue managers on z/OS»](#), à la page 930

Use these instructions to configure queue managers on IBM MQ for z/OS.

Création de gestionnaires de files d'attente sur Multiplatforms

Avant de pouvoir utiliser des messages et des files d'attente, vous devez créer et démarrer au moins un gestionnaire de files d'attente et ses objets associés. Un gestionnaire de files d'attente gère les

ressources qui lui sont associées, en particulier les files d'attente qu'il possède. Il fournit des services de mise en file d'attente aux applications pour les appels et les commandes MQI (Message queuing Interface) permettant de créer, de modifier, d'afficher et de supprimer des objets IBM MQ.

Avant de commencer

Important : IBM MQ ne prend pas en charge les noms de machine qui contiennent des espaces. Si vous installez IBM MQ sur un ordinateur dont le nom de machine contient des espaces, vous ne pouvez pas créer de gestionnaires de files d'attente.

Avant de créer un gestionnaire de files d'attente, vous devez tenir compte de plusieurs points, en particulier dans un environnement de production. Utilisez la liste de contrôle suivante:

Installation associée au gestionnaire de files d'attente

Pour créer un gestionnaire de files d'attente, utilisez la IBM MQ commande de contrôle `crtmqm`.

La commande `crtmqm` associe automatiquement un gestionnaire de files d'attente à l'installation à partir de laquelle la commande `crtmqm` a été émise. Pour les commandes qui fonctionnent sur un gestionnaire de files d'attente, vous devez exécuter la commande à partir de l'installation associée au gestionnaire de files d'attente. Vous pouvez modifier l'installation associée d'un gestionnaire de files d'attente à l'aide de la commande `setmqm`. Notez que le programme d'installation d' Windows n'ajoute pas l'utilisateur qui effectue l'installation au groupe mqm. Pour plus de détails, voir [Droits d'administration IBM MQ sur AIX, Linux®, and Windows](#).

Conventions de dénomination

Utilisez des noms en majuscules afin de pouvoir communiquer avec d'autres gestionnaires de files d'attente sur toutes les plateformes. N'oubliez pas que les noms sont attribués exactement au fur et à mesure que vous les entrez. Pour éviter les inconvénients d'un grand nombre de dactylographie, n'utilisez pas de noms trop longs.

Indiquez un nom de gestionnaire de files d'attente unique

Lorsque vous créez un gestionnaire de files d'attente, assurez-vous qu'aucun autre gestionnaire de files d'attente n'a le même nom dans votre réseau. Les noms de gestionnaire de files d'attente ne sont pas vérifiés lors de la création du gestionnaire de files d'attente et les noms qui ne sont pas uniques vous empêchent de créer des canaux pour la mise en file d'attente répartie. De plus, si vous utilisez le réseau pour la messagerie de publication / abonnement, les abonnements sont associés au nom du gestionnaire de files d'attente qui les a créés. Par conséquent, si les gestionnaires de files d'attente du cluster ou de la hiérarchie portent le même nom, les publications peuvent ne pas les atteindre.

Une façon de garantir l'unicité consiste à préfixer chaque nom de gestionnaire de files d'attente avec son propre nom de noeud unique. Par exemple, si un noeud est appelé ACCOUNTS, vous pouvez nommer votre gestionnaire de files d'attente ACCOUNTS.SATURN.QUEUE.MANAGER, où SATURN identifie un gestionnaire de files d'attente particulier et QUEUE.MANAGER est une extension que vous pouvez attribuer à tous les gestionnaires de files d'attente. Vous pouvez également l'omettre, mais notez que ACCOUNTS.SATURN et ACCOUNTS.SATURN.QUEUE.MANAGER sont des noms de gestionnaire de files d'attente différents.

Si vous utilisez IBM MQ pour la communication avec d'autres entreprises, vous pouvez également inclure votre propre nom d'entreprise comme préfixe. Ceci n'est pas illustré dans les exemples, car cela les rend plus difficiles à suivre.

Remarque : Les noms de gestionnaire de files d'attente dans les commandes de contrôle sont sensibles à la casse. Cela signifie que vous êtes autorisé à créer deux gestionnaires de files d'attente avec les noms `jupiter.queue.manager` et `JUPITER.queue.manager`. Cependant, il est préférable d'éviter de telles complications.

Limitation du nombre de gestionnaires de files d'attente

Vous pouvez créer autant de gestionnaires de files d'attente que les ressources le permettent. Toutefois, étant donné que chaque gestionnaire de files d'attente a besoin de ses propres ressources, il est généralement préférable d'avoir un gestionnaire de files d'attente avec 100 files d'attente sur un noeud plutôt que dix gestionnaires de files d'attente avec dix files d'attente chacun.

Dans les systèmes de production, de nombreux processeurs peuvent être exploités avec un seul gestionnaire de files d'attente, mais les machines serveur de plus grande taille peuvent s'exécuter plus efficacement avec plusieurs gestionnaires de files d'attente.

Indiquez un gestionnaire de files d'attente par défaut

Chaque noeud doit disposer d'un gestionnaire de files d'attente par défaut, bien qu'il soit possible de configurer IBM MQ sur un noeud sans gestionnaire de files d'attente. Le gestionnaire de files d'attente par défaut est le gestionnaire de files d'attente auquel les applications se connectent si elles ne spécifient pas de nom de gestionnaire de files d'attente dans un appel MQCONN . C'est également le gestionnaire de files d'attente qui traite les commandes MQSC lorsque vous appelez la commande runmqsc sans spécifier de nom de gestionnaire de files d'attente.

Le gestionnaire de files d'attente désigné comme gestionnaire de files d'attente par défaut remplace toute autre spécification de gestionnaire de files d'attente par défaut pour le noeud considéré.

La modification de la gestion de file d'attente par défaut peut affecter d'autres utilisateurs ou applications. La modification n'a aucun effet sur les applications actuellement connectées, car elles peuvent utiliser le descripteur de leur appel de connexion d'origine dans tous les autres appels MQI. Ce descripteur garantit que les appels sont dirigés vers le même gestionnaire de files d'attente. Toutes les applications qui se connectent *après* avoir modifié le gestionnaire de files d'attente par défaut se connectent au nouveau gestionnaire de files d'attente par défaut. C'est peut-être ce que vous souhaitez, mais vous devez en tenir compte avant de modifier la valeur par défaut.

La création d'un gestionnaire de files d'attente par défaut est décrite dans [«Création d'un gestionnaire de files d'attente par défaut»](#), à la page 12.

Indiquez une file d'attente de rebut

La file d'attente de rebut est une file d'attente locale dans laquelle les messages sont insérés s'ils ne peuvent pas être acheminés vers leur destination prévue.

Il est indispensable de définir une file d'attente de rebut dans chaque gestionnaire de files d'attente de votre réseau. A défaut, les erreurs des programmes d'application risquent de provoquer la fermeture des canaux et vous ne recevrez pas les réponses des commandes administratives.

Par exemple, si une application tente d'insérer un message dans une file d'attente d'un autre gestionnaire de files d'attente, mais qu'elle indique un nom de file d'attente incorrect, le canal est arrêté et le message reste dans la file d'attente de transmission. Les autres applications ne peuvent alors pas utiliser ce canal pour leurs messages.

Les canaux ne sont pas affectés si les gestionnaires de files d'attente possèdent des files d'attente de rebut. Le message non distribué est inséré dans la file d'attente de rebut à la réception, laissant le canal et sa file d'attente de transmission disponibles.

Lorsque vous créez un gestionnaire de files d'attente, utilisez l'indicateur **-u** pour spécifier le nom de la file d'attente de rebut. Vous pouvez également utiliser une commande MQSC pour modifier les attributs d'un gestionnaire de files d'attente que vous avez déjà défini pour spécifier la file d'attente de rebut à utiliser. Pour obtenir un exemple de la commande MQSC ALTER, voir [Affichage et modification des attributs de gestionnaire de files d'attente](#) .

Indiquez une file d'attente de transmission par défaut

Une file d'attente de transmission est une file d'attente locale dans laquelle les messages en transit vers un gestionnaire de files d'attente éloignées sont mis en file d'attente avant la transmission. La file d'attente de transmission par défaut est la file d'attente utilisée en l'absence de file d'attente de transmission explicitement définie. Une file d'attente de transmission par défaut peut être affectée à chaque gestionnaire de files d'attente.

Lorsque vous créez un gestionnaire de files d'attente, utilisez l'indicateur **-d** pour spécifier le nom de la file d'attente de transmission par défaut. Cela ne crée pas réellement la file d'attente ; vous devez le faire explicitement ultérieurement. Pour plus d'informations, voir [Utilisation des files d'attente locales](#) .

Spécifiez les paramètres de journalisation dont vous avez besoin

Vous pouvez spécifier des paramètres de journalisation dans la commande crtmqm , y compris le type de journalisation, ainsi que le chemin et la taille des fichiers journaux.

Dans un environnement de développement, les paramètres de consignation par défaut doivent être adéquats. Toutefois, vous pouvez modifier les valeurs par défaut si, par exemple:

- Vous disposez d'une configuration de système bas de gamme qui ne prend pas en charge les journaux volumineux.
- Vous prévoyez un grand nombre de messages longs dans vos files d'attente en même temps.
- Vous anticipez un grand nombre de messages persistants qui transitent par le gestionnaire de files d'attente.

Une fois que vous avez défini les paramètres de journalisation, certains d'entre eux ne peuvent être modifiés qu'en supprimant le gestionnaire de files d'attente et en le recréant avec le même nom mais avec des paramètres de journalisation différents.

Pour plus d'informations sur les paramètres de journalisation, voir [«Consignation: S'assurer que les messages ne sont pas perdus»](#), à la page 703.

AIX

Pour les systèmes IBM MQ for UNIX uniquement

Vous pouvez créer le répertoire de gestionnaire de files d'attente `/var/mqm/qmgrs/qmgr`, même sur un système de fichiers local distinct, avant d'utiliser la commande `crtmqm`. Lorsque vous utilisez `crtmqm`, si le répertoire `/var/mqm/qmgrs/qmgr` existe, est vide et appartient à `mqm`, il est utilisé pour les données du gestionnaire de files d'attente. Si le répertoire n'appartient pas à `mqm`, la création échoue avec un First Failure Support Technology (FFST) message. Si le répertoire n'est pas vide, un nouveau répertoire est créé.

Pourquoi et quand exécuter cette tâche

Pour créer un gestionnaire de files d'attente, utilisez la IBM MQ commande de contrôle `crtmqm`. Pour plus d'informations, voir `crtmqm`. La commande `crtmqm` crée automatiquement les objets par défaut et les objets système requis (voir Objets par défaut du système). Les objets par défaut forment la base de toutes les définitions d'objet que vous créez ; les objets système sont requis pour le fonctionnement du gestionnaire de files d'attente.

Windows

Sur les systèmes Windows, vous avez la possibilité de démarrer plusieurs instances du gestionnaire de files d'attente à l'aide de l'option `sax` de la commande `crtmqm`.

Une fois que vous avez créé un gestionnaire de files d'attente et ses objets, vous pouvez utiliser la commande `strmqm` pour démarrer le gestionnaire de files d'attente.

Procédure

- Pour plus d'informations sur la création et la gestion des gestionnaires de files d'attente, voir les sous-rubriques suivantes:
 - [«Création d'un gestionnaire de files d'attente par défaut»](#), à la page 12
 - [«Faire d'un gestionnaire de files d'attente existant la valeur par défaut»](#), à la page 14
 - [«Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente»](#), à la page 15

Concepts associés

[Utilisation des gestionnaires de files d'attente](#)

Tâches associées

[Création du gestionnaire de files d'attente QM1](#)

[«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms»](#), à la page 96

Vous pouvez modifier le comportement d'IBM MQ ou d'un gestionnaire de files d'attente individuel en fonction des besoins de votre installation en éditant les informations dans les fichiers de configuration (`.ini`). Vous pouvez également modifier les options de configuration pour IBM MQ MQI clients.

[«Configuring queue managers on z/OS»](#), à la page 930

Use these instructions to configure queue managers on IBM MQ for z/OS.

Référence associée

[Objets système et par défaut](#)

[crtmqm](#)

Linux

AIX

Répertoire éphémère configurable

Le répertoire éphémère configurable définit l'emplacement auquel les données éphémères du gestionnaire de files d'attente doivent se trouver. Il peut être utilisé pour permettre aux sockets de domaine AIX and Linux d'être placés sur un système de fichiers non monté dans un environnement Red Hat® OpenShift®.

Avant IBM MQ 9.2.0, sur les plateformes AIX and Linux, lorsqu'un gestionnaire de files d'attente est en cours d'exécution, des sockets de domaine AIX and Linux sont créés dans le répertoire `/var/mqm/sockets`. Lors de l'exécution du gestionnaire de files d'attente à l'intérieur d'un conteneur, avec `/var/mqm` comme système de fichiers monté, certaines plateformes Linux peuvent empêcher la création de ces sockets de domaine, car elles permettent à certains processus de l'extérieur du conteneur d'interférer avec les opérations à l'intérieur du conteneur. Ce problème empêche IBM MQ de s'exécuter dans une plateforme de conteneur Red Hat OpenShift, dans le contexte de sécurité par défaut.

Depuis la IBM MQ 9.2.0, l'attribut **EphemeralPrefix** peut être utilisé pour configurer l'emplacement du répertoire éphémère. Si vous n'utilisez pas cet attribut, vous ne verrez aucun changement de comportement.

Lorsqu'une entrée de gestionnaire de files d'attente est créée dans `mqm.ini` (à l'aide des commandes **crtmqm** ou **addmqinf**), l'attribut **EphemeralPrefix** est ajouté si vous:

- Définissez l'attribut **DefaultEphemeralPrefix** dans «[Strophe AllQueueManagers du fichier mqm.ini](#)», à la page 102.
- Définissez la variable d'environnement **MQ_EPHEMERAL_PREFIX**.
- Spécifiez **-v EphemeralPrefix** pour la commande **addmqinf** uniquement.

Vous pouvez également ajouter explicitement l'attribut **EphemeralPrefix** à un gestionnaire de files d'attente existant lorsqu'il est arrêté, et il est ajouté lorsque le gestionnaire de files d'attente est redémarré.

Si vous spécifiez un attribut **EphemeralPrefix**, lorsque le gestionnaire de files d'attente démarre, les données éphémères du gestionnaire de files d'attente sont créées sous ce préfixe et non à son emplacement habituel. A savoir :

- Les fichiers de socket qui sont généralement présents sous `/var/mqm/sockets/<QM>` se trouvent désormais sous `/EphemeralPrefix/sockets/<QM>`
- Les fichiers de sous-portion de mémoire qui sont généralement présents sous `/Prefix/qmgrs/<QM>/@<Subpool>` se trouvent désormais sous `/EphemeralPrefix/qmgrs/<QM>/@<Subpool>`

Remarques :

- `/var/mqm/sockets/@SYSTEM` reste à son emplacement fixe et ne fait pas partie de l'attribut **EphemeralPrefix**.
- `AMQCLCHL.TAB` reste sous `/Prefix/qmgrs/<QM>/@ipcc` et ne fait pas partie de l'attribut **EphemeralPrefix**.

Le nombre de caractères que l'attribut **EphemeralPrefix** peut inclure dépend de votre plateforme:

- **Linux** **AIX** Sur les plateformes AIX and Linux, il est limité à 12 caractères.
- **IBM i** Sous IBM i, il est limité à 24 caractères.

Si vous spécifiez un attribut **EphemeralPrefix** qui est trop long ou qui n'existe pas, vous recevez un message [AMQ7001E](#) :

AMQ7001E: L'emplacement spécifié pour le gestionnaire de files d'attente n'est pas valide

Multi Répertoire userdata

Vous pouvez utiliser le répertoire `userdata` pour stocker le statut des applications persistantes.

Chaque gestionnaire de files d'attente IBM MQ dispose d'un système de fichiers dédié pour son état persistant, qui inclut à la fois ses données de file d'attente et son journal de reprise. Le système de fichiers inclut un répertoire `userdata` que vous pouvez utiliser pour stocker les informations de statut persistantes de vos applications. Voir [Contenu de l'annuaire sur les systèmes Unix et Linux](#) et [Contenu de l'annuaire sur les systèmes Windows](#).

Le répertoire `userdata` peut être utile dans un certain nombre de situations, par exemple:

- Dans les configurations de gestionnaire de files d'attente de données répliquées (RDQM), les informations d'application sont également déplacées lorsqu'un gestionnaire de files d'attente bascule vers un autre noeud (voir «[Stockage du statut de l'application persistante](#)», à la page 635).
- Pour les gestionnaires de files d'attente multi-instance afin que leur état d'application soit localisé avec leurs données de gestionnaire de files d'attente sur le système de fichiers du réseau partagé.
- Plus généralement, où les applications sont configurées pour les services de gestionnaire de files d'attente.

Si vous choisissez de stocker l'état de l'application dans le répertoire `userdata`, vous devez savoir que les données écrites à cet emplacement peuvent consommer l'espace disque disponible alloué au gestionnaire de files d'attente. Vous devez vous assurer que l'espace disque disponible reste suffisant pour que le gestionnaire de files d'attente puisse écrire des données de file d'attente, des journaux et d'autres informations d'état persistantes.

Le répertoire `userdata` est propriétaire de l'utilisateur et du groupe `mqm` et est accessible en lecture à tous, de sorte que les utilisateurs puissent y accéder sans avoir à faire partie du groupe d'administrateurs IBM MQ (c'est-à-dire `mqm`). Vous ne pouvez pas modifier les droits du répertoire `userdata`, mais vous pouvez y créer du contenu avec la propriété et les droits dont vous avez besoin.

Multi Création d'un gestionnaire de files d'attente par défaut

Le gestionnaire de files d'attente par défaut est le gestionnaire de files d'attente auquel les applications se connectent si elles ne spécifient pas de nom de gestionnaire de files d'attente dans un appel `MQCONN`. C'est également le gestionnaire de files d'attente qui traite les commandes `MQSC` lorsque vous appelez la commande `runmqsc` sans spécifier de nom de gestionnaire de files d'attente. Pour créer un gestionnaire de files d'attente, utilisez la IBM MQ commande de contrôle `crtmqm`.

Avant de commencer

Avant de créer un gestionnaire de files d'attente par défaut, lisez les remarques décrites dans «[Création de gestionnaires de files d'attente sur Multiplatforms](#)», à la page 7.

Linux **AIX** Lorsque vous utilisez `crtmqm` pour créer un gestionnaire de files d'attente sous AIX and Linux, si le répertoire `/var/mqm/qmgrs/qmgr` existe déjà, qu'il appartient à `mqm` et qu'il est vide, il est utilisé pour les données du gestionnaire de files d'attente. Si le répertoire n'appartient pas à `mqm`, la création du gestionnaire de files d'attente échoue avec un message First Failure Support Technology (FFST). Si le répertoire n'est pas vide, un nouveau répertoire est créé pour les données du gestionnaire de files d'attente.

Cette remarque s'applique même lorsque le répertoire `/var/mqm/qmgrs/qmgr` existe déjà sur un système de fichiers local distinct.

Pourquoi et quand exécuter cette tâche

Lorsque vous créez un gestionnaire de files d'attente à l'aide de la commande `crtmqm`, la commande crée automatiquement les objets par défaut et les objets système requis. Les objets par défaut forment

la base de toutes les définitions d'objet que vous créez et les objets système sont requis pour le fonctionnement du gestionnaire de files d'attente.

En incluant les paramètres appropriés dans la commande, vous pouvez également définir, par exemple, le nom de la file d'attente de transmission par défaut à utiliser par le gestionnaire de files d'attente et le nom de la file d'attente de rebut.

Windows Sous Windows, vous pouvez utiliser l'option **sax** de la commande **crtmqm** pour démarrer plusieurs instances du gestionnaire de files d'attente.

Pour plus d'informations sur la commande **crtmqm** et sa syntaxe, voir **crtmqm**.

Procédure

- Pour créer un gestionnaire de files d'attente par défaut, utilisez la commande **crtmqm** avec l'indicateur **-q**.

L'exemple suivant de la commande **crtmqm** crée un gestionnaire de files d'attente par défaut appelé **SATURN.QUEUE.MANAGER**:

```
crtmqm -q -d MY.DEFAULT.XMIT.QUEUE -u SYSTEM.DEAD.LETTER.QUEUE SATURN.QUEUE.MANAGER
```

où :

-q

Indique que ce gestionnaire de files d'attente est le gestionnaire de files d'attente par défaut.

-d MY.DEFAULT.XMIT.QUEUE

Nom de la file d'attente de transmission par défaut à utiliser par ce gestionnaire de files d'attente.

Remarque : IBM MQ ne crée pas de file d'attente de transmission par défaut pour vous ; vous devez la définir vous-même.

-u SYSTEM.DEAD.LETTER.QUEUE

Nom de la file d'attente de rebut par défaut créée par IBM MQ lors de l'installation.

SATURN.QUEUE.MANAGER

Nom de ce gestionnaire de files d'attente. Il doit s'agir du dernier paramètre spécifié dans la commande **crtmqm**.

Que faire ensuite

Une fois que vous avez créé un gestionnaire de files d'attente et ses objets, utilisez la commande **strmqm** pour démarrer le gestionnaire de files d'attente.

Concepts associés

Utilisation des files d'attente locales

Tâches associées

«Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente», à la page 15

Les informations de configuration IBM MQ sont stockées dans des fichiers de configuration sous AIX, Linux, and Windows. Après avoir créé un gestionnaire de files d'attente, sauvegardez vos fichiers de configuration. Ensuite, si vous créez un autre gestionnaire de files d'attente à l'origine des problèmes, vous pouvez rétablir les sauvegardes une fois que vous avez supprimé la source du problème.

Affichage et modification des attributs du gestionnaire de files d'attente

Référence associée

Objets système et par défaut

Faire d'un gestionnaire de files d'attente existant la valeur par défaut

Vous pouvez faire d'un gestionnaire de files d'attente existant le gestionnaire de files d'attente par défaut manuellement à l'aide d'un éditeur de texte ou, sous Windows et Linux, à l'aide de IBM MQ Explorer.

Pourquoi et quand exécuter cette tâche

Pour utiliser un éditeur de texte afin de faire d'un gestionnaire de files d'attente existant le gestionnaire de files d'attente par défaut, procédez comme suit.

Windows **Linux** Sur les systèmes Windows et Linux (plateformes x86 et x86-64), si vous préférez utiliser IBM MQ Explorer pour effectuer cette modification, voir [«Utilisation de IBM MQ Explorer pour définir un gestionnaire de files d'attente comme gestionnaire de files d'attente par défaut»](#), à la page 14.

Lorsque vous créez un gestionnaire de files d'attente par défaut, son nom est inséré dans l'attribut Name de la section `DefaultQueueManager` du fichier de configuration IBM MQ (`mqs.ini`). La section et son contenu sont automatiquement créés s'ils n'existent pas.

Procédure

- Pour définir un gestionnaire de files d'attente existant comme gestionnaire de files d'attente par défaut, remplacez le nom du gestionnaire de files d'attente dans l'attribut Name par le nom du nouveau gestionnaire de files d'attente par défaut. Vous pouvez le faire manuellement à l'aide d'un éditeur de texte.
- Si vous n'avez pas de gestionnaire de files d'attente par défaut sur le noeud et que vous souhaitez définir un gestionnaire de files d'attente existant comme gestionnaire de files d'attente par défaut, créez vous-même la strophe `DefaultQueueManager` avec le nom requis.
- Si vous définissez par erreur un autre gestionnaire de files d'attente comme gestionnaire de files d'attente par défaut et que vous souhaitez rétablir le gestionnaire de files d'attente par défaut d'origine, éditez la section `DefaultQueueManager` dans `mqs.ini`, en remplaçant le gestionnaire de files d'attente par défaut non souhaité par celui de votre choix.

Tâches associées

[«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms»](#), à la page 96

Vous pouvez modifier le comportement d'IBM MQ ou d'un gestionnaire de files d'attente individuel en fonction des besoins de votre installation en éditant les informations dans les fichiers de configuration (`.ini`). Vous pouvez également modifier les options de configuration pour IBM MQ MQI clients.

Windows **Linux** Utilisation de IBM MQ Explorer pour définir un gestionnaire de files d'attente comme gestionnaire de files d'attente par défaut

Sur les systèmes Windows et Linux (plateformes x86 et x86-64), vous pouvez utiliser IBM MQ Explorer pour faire d'un gestionnaire de files d'attente existant le gestionnaire de files d'attente par défaut.

Pourquoi et quand exécuter cette tâche

Pour utiliser IBM MQ Explorer afin de faire d'un gestionnaire de files d'attente existant le gestionnaire de files d'attente par défaut sur les systèmes Windows et Linux (x86 et x86-64), procédez comme suit.

Si vous préférez utiliser un éditeur de texte pour effectuer cette modification manuellement, voir [«Faire d'un gestionnaire de files d'attente existant la valeur par défaut»](#), à la page 14.

Procédure

1. Ouvrez IBM MQ Explorer.
2. Cliquez avec le bouton droit de la souris sur **IBM MQ**, puis sélectionnez **Propriétés ...**. Le panneau **Propriétés d' IBM MQ** s'affiche.
3. Entrez le nom du gestionnaire de files d'attente par défaut dans la zone **Nom du gestionnaire de files d'attente par défaut**.
4. Cliquez sur **OK**.

Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente

Les informations de configuration IBM MQ sont stockées dans des fichiers de configuration sous AIX, Linux, and Windows. Après avoir créé un gestionnaire de files d'attente, sauvegardez vos fichiers de configuration. Ensuite, si vous créez un autre gestionnaire de files d'attente à l'origine des problèmes, vous pouvez rétablir les sauvegardes une fois que vous avez supprimé la source du problème.



Pourquoi et quand exécuter cette tâche

En règle générale, sauvegardez vos fichiers de configuration chaque fois que vous créez un gestionnaire de files d'attente.

Il existe deux types de fichier de configuration:

- Lorsque vous installez le produit, le fichier de configuration IBM MQ (`mqs.ini`) est créé. Il contient une liste de gestionnaires de files d'attente qui est mise à jour chaque fois que vous créez ou supprimez un gestionnaire de files d'attente. Il existe un fichier `mqs.ini` par noeud.
- Lorsque vous créez un gestionnaire de files d'attente, un nouveau fichier de configuration de gestionnaire de files d'attente (`qm.ini`) est automatiquement créé. Contient les paramètres de configuration du gestionnaire de files d'attente.

Si vous avez installé le service AMQP, vous devez sauvegarder un fichier de configuration supplémentaire:

-  Sur les systèmes Windows : `amqp_win.properties`
-  Sur les systèmes AIX and Linux : `amqp_unix.properties`

Tâches associées

[«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms», à la page 96](#)

Vous pouvez modifier le comportement d' IBM MQ ou d'un gestionnaire de files d'attente individuel en fonction des besoins de votre installation en éditant les informations dans les fichiers de configuration (`.ini`). Vous pouvez également modifier les options de configuration pour IBM MQ MQI clients.

[«Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ», à la page 737](#)

Vous pouvez protéger les gestionnaires de files d'attente contre les éventuelles altérations causées par des pannes matérielles en sauvegardant les gestionnaires de files d'attente et les données de gestionnaire de files d'attente, en sauvegardant uniquement la configuration du gestionnaire de files d'attente et en utilisant un gestionnaire de files d'attente de sauvegarde.

Déplacement d'un gestionnaire de files d'attente sur un système d'exploitation différent

Suivez les instructions ci-dessous pour déplacer un gestionnaire de files d'attente d'un système d'exploitation vers un autre. Notez qu'il ne s'agit **pas** d'une migration de gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Vous déplacez un gestionnaire de files d'attente en le recréant sur le système cible. La procédure recrée la configuration du gestionnaire de files d'attente ; elle ne tente pas de recréer l'état en cours du gestionnaire de files d'attente, par exemple en déchargeant et en rechargeant les files d'attente.

Procédure

1. Connectez-vous au système source en tant qu'utilisateur dans le groupe d'administrateurs IBM MQ (mqm).
2. Sauvegardez les informations de configuration du gestionnaire de files d'attente à déplacer avec la commande suivante :




```
dmpmqcfg -a -m QM_name > QM_file
```

où :

- *nom_gest_files_attente* est le nom du gestionnaire de files d'attente à déplacer.
- *fichier_gest_files_attente* est le nom et le chemin d'accès d'un fichier local sur le système source dans lequel sont écrites les informations de configuration.

Voir **dmpmqcfg** pour plus d'informations.

3. Si le gestionnaire de files d'attente fait partie d'une configuration répartie, mettez-le au repos. Assurez-vous qu'aucun message n'est en cours de traitement, puis arrêtez le gestionnaire de files d'attente.
4. Si vous procédez au déplacement d'une version du produit vers une autre, migrez le gestionnaire de files d'attente qui se trouve sur votre système d'exploitation actuel vers la version la plus récente. Si votre système d'exploitation actuel est :

-  Windows, voir [Migration d'un gestionnaire de files d'attente sous Windows](#)
-  Linux  AIX ou Linux, voir [Migration d'un gestionnaire de files d'attente sous AIX and Linux](#)

Vérifiez que vos applications existantes fonctionnent toujours.

5. Créez un gestionnaire de files d'attente vide sur votre nouveau système d'exploitation avec la commande **crtmqm**.
6. Copiez vos définitions d'objet dans le nouveau gestionnaire de files d'attente avec la commande **dmpmqcfg**.

Soyez très attentif lorsque vous copiez les définitions d'objet car il peut être nécessaire de modifier manuellement des définitions :

- Vous devez vérifier divers attributs pouvant requérir des modifications, Ces gestionnaires sont les suivants :
 - Les adresses IP et les ports pour les canaux, les programmes d'écoute et d'autres objets
 - Les informations de sécurité telles que les ID utilisateur
 - **startcmd** dans les services
 - Divers autres attributs
- Les abonnés durables qui ne sont pas administrés peuvent perdre des messages.
- D'autres gestionnaires de files d'attente peuvent également nécessiter des modifications pour que leurs canaux puissent se connecter au gestionnaire de files d'attente déplacé.

Une fois les définitions copiées, vous devez copier les messages d'application du gestionnaire de files d'attente sur votre système d'exploitation d'origine vers le gestionnaire de files d'attente sur votre nouveau système d'exploitation, à l'aide de l'application qui déplace les messages. Ensuite, vérifiez que vos applications fonctionnent toujours.

Configuration des connexions entre le client et le serveur

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

Pourquoi et quand exécuter cette tâche

Dans IBM MQ, les liaisons de communication logiques entre les objets sont appelées *canaux*. Les canaux utilisés pour connecter IBM MQ MQI clients aux serveurs sont appelés canaux MQI. Vous configurez des définitions de canal à chaque extrémité de votre lien afin que votre application IBM MQ sur le IBM MQ MQI client puisse communiquer avec le gestionnaire de files d'attente sur le serveur.

Avant de définir vos canaux MQI, vous devez décider de la forme de communication que vous allez utiliser et définir la connexion à chaque extrémité du canal.

Si vous définissez un canal MQI entre un IBM MQ MQI client et un gestionnaire de files d'attente qui se trouvent sur des réseaux physiques différents ou qui communiquent via un pare-feu, l'utilisation de IBM MQ Internet Pass-Thru peut simplifier la configuration. Pour plus d'informations, voir [IBM MQ Internet Pass-Thru](#).

Procédure

1. Déterminez la forme de communication que vous allez utiliser.
Voir «Type de communication à utiliser», à la page 17.
2. Définissez la connexion à chaque extrémité du canal.
Pour définir la connexion, vous devez:
 - a) Configurez la connexion.
 - b) Enregistrez les valeurs des paramètres dont vous avez besoin pour les définitions de canal.
 - c) Activez le serveur pour détecter les demandes réseau entrantes provenant de votre IBM MQ MQI client, en démarrant un *programme d'écoute*.

Concepts associés

«Fichier de configuration IBM MQ MQI client , mqclient.ini», à la page 174

Vous pouvez configurer vos clients en utilisant des attributs dans un fichier texte. Ces attributs peuvent être remplacés par des variables d'environnement ou par d'autres méthodes spécifiques à la plateforme.

Tâches associées

«Utilisation des variables d'environnement IBM MQ», à la page 67

Vous pouvez utiliser des commandes pour afficher les paramètres en cours ou pour réinitialiser les valeurs des variables d'environnement IBM MQ .

[Connexion des applications client IBM MQ MQI aux gestionnaires de files d'attente](#)

Référence associée

[AFFICHER CHLAUTH](#)

[SET CHLAUTH](#)

Type de communication à utiliser

Différentes plateformes prennent en charge différents protocoles de communication. Votre choix de protocole de transmission dépend de votre combinaison de IBM MQ MQI client et de plateformes serveur.

Types de protocole de transmission pour les canaux MQI










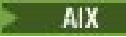











En fonction de vos plateformes client et serveur, il existe jusqu'à quatre types de protocole de transmission pour les canaux MQI:

- TCP/IP


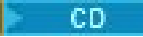
- LU 6.2
- NetBIOS
- SPX

Lorsque vous définissez vos canaux MQI, chaque définition de canal doit spécifier un attribut de protocole de transmission (type de transport). Un serveur n'étant pas limité à un seul protocole, des définitions de canal différentes peuvent spécifier des protocoles différents. Pour IBM MQ MQI clients, il peut être utile de disposer de canaux MQI alternatifs utilisant des protocoles de transmission différents.

Votre choix de protocole de transmission dépend également de votre combinaison particulière de plateformes client et serveur IBM MQ . Les combinaisons possibles sont indiquées dans le tableau suivant.

Protocole de transmission	IBM MQ MQI client	IBM MQServeur
TCP/IP «1», à la page 18	 IBM i  AIX  Linux  Windows	 IBM i  AIX  Linux  Windows  z/OS
LU 6.2	 AIX  Linux«2», à la page 18  Windows	 IBM i  AIX  Linux«2», à la page 18  Windows  z/OS
NetBIOS	 Windows	 Windows
SPX	 Windows	 Windows

Remarques :

1.   Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).
2. Sauf Linux (plateforme POWER)

Concepts associés

«Définition d'une connexion TCP sous Windows», à la page 280

Définissez une connexion TCP en configurant un canal à l'extrémité émettrice pour spécifier l'adresse de la cible et en exécutant un programme d'écoute à l'extrémité réceptrice.

«Définition d'une connexion TCP sous AIX and Linux», à la page 288

La définition de canal à l'extrémité émettrice indique l'adresse de la cible. Le programme d'écoute ou le démon inet est configuré pour la connexion à l'extrémité réceptrice.

«Définition d'une connexion TCP sous IBM i», à la page 308

Vous pouvez définir une connexion TCP dans la définition de canal à l'aide de la zone Nom de connexion.

«Defining a TCP connection on z/OS», à la page 1030

To define a TCP connection, there are a number of settings to configure.

«Définition d'une connexion LU 6.2 sous Windows», à la page 282

SNA doit être configuré pour qu'une conversation LU 6.2 puisse être établie entre les deux machines.

«Définition d'une connexion LU 6.2 sous AIX and Linux», à la page 292

SNA doit être configuré pour qu'une conversation LU 6.2 puisse être établie entre les deux machines.

«Définition d'une connexion LU 6.2 sous IBM i», à la page 310

Définissez les détails des communications LU 6.2 à l'aide d'un nom de mode, d'un nom TP et d'un nom de connexion d'une connexion LU 6.2 complète.

«Définition d'une connexion NetBIOS sous Windows», à la page 284

Une connexion NetBIOS s'applique uniquement à un client et un serveur exécutant Windows. IBM MQ utilise trois types de ressource NetBIOS lors de l'établissement d'une connexion NetBIOS à un autre produit IBM MQ : les sessions, les commandes et les noms. Chacune de ces ressources a une limite, qui est établie par défaut ou par choix lors de l'installation de NetBIOS.

Tâches associées

«Définition d'une connexion Aspera gateway sur les plateformes Linux ou Windows», à la page 920

IBM Aspera faspio Gateway fournit un tunnel TCP/IP rapide qui peut augmenter considérablement le débit du réseau pour IBM MQ. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows.

Référence associée

«Nombre maximal de connexions TCP/IP», à la page 19

Le nombre de demandes de connexion en attente pouvant être mises en file d'attente sur un seul port TCP/IP dépend de la plateforme. Une erreur se produit si la limite est atteinte.

«Defining an LU6.2 connection for z/OS using APPC/MVS», à la page 1033

To define an LU6.2 connection there are a number of settings to configure.

Nombre maximal de connexions TCP/IP

Le nombre de demandes de connexion en attente pouvant être mises en file d'attente sur un seul port TCP/IP dépend de la plateforme. Une erreur se produit si la limite est atteinte.

Cette limite de connexion n'est pas identique au nombre maximal de clients que vous pouvez connecter à un serveur IBM MQ . Vous pouvez connecter davantage de clients à un serveur, jusqu'au niveau déterminé par les ressources système du serveur. Les valeurs de journal des éléments en attente pour les demandes de connexion sont répertoriées dans le tableau suivant:







Plateforme serveur	Nombre maximal de demandes de connexion
 AIX	100
 Linux	100
 IBM i	255
Serveur  Windows	100

Tableau 2. Nombre maximal de demandes de connexion en attente mises en file d'attente sur un port TCP/IP (suite)

Plateforme serveur	Nombre maximal de demandes de connexion
Poste de travail  Windows	100
 z/OS	255

Si la limite de connexion est atteinte, le client reçoit le code retour MQRC_HOST_NOT_AVAILABLE de l'appel MQCONN et une erreur AMQ9202 dans le journal des erreurs du client (/var/mqm/errors/AMQERR0n.LOG sur les systèmes AIX and Linux ou amqerr0n.log dans le sous-répertoire des erreurs de l'installation du client IBM MQ sous Windows). Si le client relance la demande MQCONN, il se peut qu'elle aboutisse.

Pour augmenter le nombre de demandes de connexion que vous pouvez effectuer et éviter que des messages d'erreur ne soient générés par cette limitation, vous pouvez disposer de plusieurs programmes d'écoute chacun écoutant sur un port différent ou disposer de plusieurs gestionnaires de files d'attente.





Comment configurer un IBM MQ MQI client

Pour configurer un client, suivez les instructions ci-dessous.

Avant de commencer

Pour configurer un IBM MQ MQI client, vous devez avoir installé un serveur IBM MQ opérationnel auquel le client se connectera.

Procédure

1. Vérifiez que vous disposez d'une plateforme appropriée pour un client IBM MQ MQI et que le matériel et les logiciels répondent aux exigences.
La prise en charge de la plateforme est décrite dans [Prise en charge de la plateforme pour les clients IBM MQ](#).
2. Décidez de la manière dont vous allez installer IBM MQ sur votre poste de travail client, puis suivez les instructions correspondant à votre combinaison particulière de plateformes client et serveur.
L'installation est décrite dans les rubriques suivantes :
 -  [Installation d'un client IBM MQ sous AIX](#)
 -  [Installation d'un client IBM MQ sous Linux](#)
 -  [Installation d'un client IBM MQ sous Windows](#)
 -  [Installation d'un client IBM MQ sous IBM i](#)
3. Assurez-vous que vos liaisons de communication sont configurées et connectées.
La configuration des liaisons de communication est décrite dans la rubrique [Configuration des connexions entre le serveur et le client](#).
4. Vérifiez que votre installation fonctionne correctement.
Consultez la section de vérification de la procédure d'installation pour la ou les plateformes que votre entreprise utilise.
5. Une fois que vous avez vérifié l'installation du IBM MQ MQI client, déterminez si vous devez sécuriser le client.
La sécurité du client est décrite dans [Configuration de la sécurité du IBM MQ MQI client](#).
6. Configurez les canaux entre le client IBM MQ MQI et le serveur requis par les applications IBM MQ que vous souhaitez exécuter sur le client.

La configuration des canaux est décrite dans la rubrique [Définition de canaux MQI](#). Si vous utilisez TLS, vous devez prendre en compte quelques informations supplémentaires.

Ces informations sont décrites dans la rubrique [Spécification de l'utilisation de TLS par un canal MQI](#). Il peut être nécessaire d'utiliser un fichier de configuration IBM MQ MQI client ou des variables d'environnement IBM MQ pour configurer les canaux. Les variables d'environnement IBM MQ sont décrites dans [Utilisation des variables d'environnement IBM MQ](#).

7. Pour une description complète des applications IBM MQ, voir [Développement d'applications](#).
8. Lors de la conception, de la génération et de l'exécution d'applications dans l'environnement IBM MQ MQI client, vous devez prendre en compte les différences par rapport à un environnement de gestionnaire de files d'attente.

Pour plus d'informations sur ces différences, voir :

- [Utilisation de l'interface de file d'attente de messages \(MQI\) dans une application client](#)
- [Développement d'applications pour les IBM MQ MQI clients](#)
- [Connexion des applications de IBM MQ MQI client aux gestionnaires de files d'attente](#)
- [Résolution des incidents liés aux IBM MQ MQI clients](#)

Configuration d'un client transactionnel étendu

Cette collection de rubriques explique comment configurer la fonction transactionnelle étendue pour chaque catégorie du gestionnaire de transactions.

Pour chaque plateforme, le client transactionnel étendu prend en charge les gestionnaires de transactions externes suivants:

gestionnaires de transactions compatibles XA

Le client transactionnel étendu fournit l'interface du gestionnaire de ressources XA pour prendre en charge les gestionnaires de transactions compatibles XA tels que CICS et Tuxedo.

Windows Microsoft Transaction Server (systèmes Windows uniquement)

Sur les systèmes Windows uniquement, l'interface du gestionnaire de ressources XA prend également en charge Microsoft Transaction Server (MTS). Le support IBM MQ MTS fourni avec le client transactionnel étendu fournit le pont entre MTS et l'interface du gestionnaire de ressources XA.

WebSphere Application Server

WebSphere Application Server 6 et les versions ultérieures incluent un fournisseur de messagerie IBM MQ, de sorte que vous n'avez pas besoin d'utiliser le client transactionnel étendu.

Configuration de gestionnaires de transactions compatibles XA

Configurez d'abord le client de base IBM MQ, puis configurez la fonction transactionnelle étendue à l'aide des informations de ces rubriques.

Remarque : Cette section suppose que vous disposez d'une compréhension de base de l'interface XA telle que publiée par The Open Group dans *Distributed Transaction Processing: The XA Specification*.

Pour configurer un client transactionnel étendu, vous devez d'abord configurer le client de base IBM MQ comme décrit dans:

- **AIX** [Installation d'un client IBM MQ sous AIX](#)
- **Linux** [Installation d'un client IBM MQ sous Linux](#)
- **Windows** [Installation d'un client IBM MQ sous Windows](#)
- **IBM i** [Installation d'un client IBM MQ sous IBM i](#)

A l'aide des informations de votre plateforme, vous pouvez ensuite configurer la fonction transactionnelle étendue pour un gestionnaire de transactions compatible XA tel que CICS et Tuxedo.

Un gestionnaire de transactions communique avec un gestionnaire de files d'attente en tant que gestionnaire de ressources à l'aide du même canal MQI que celui utilisé par l'application client connectée au gestionnaire de files d'attente. Lorsque le gestionnaire de transactions émet un appel de fonction de gestionnaire de ressources (xa_), le canal MQI est utilisé pour transmettre l'appel au gestionnaire de files d'attente et pour recevoir la sortie du gestionnaire de files d'attente.

Le gestionnaire de transactions peut démarrer le canal MQI en émettant un appel xa_open pour ouvrir le gestionnaire de files d'attente en tant que gestionnaire de ressources, ou l'application client peut démarrer le canal MQI en émettant un appel MQCONN ou MQCONNX.

- Si le gestionnaire de transactions démarre le canal MQI et que l'application client appelle ultérieurement MQCONN ou MQCONNX sur la même unité d'exécution, l'appel MQCONN ou MQCONNX aboutit et un descripteur de connexion est renvoyé à l'application. L'application ne reçoit pas de code achèvement MQCC_WARNING avec un code anomalie MQRC_ALREADY_CONNECTED.
- Si l'application client démarre le canal MQI et que le gestionnaire de transactions appelle ultérieurement xa_open sur la même unité d'exécution, l'appel xa_open est réacheminé vers le gestionnaire de files d'attente à l'aide du canal MQI.

Dans une situation de reprise après incident, lorsqu'aucune application client n'est en cours d'exécution, le gestionnaire de transactions peut utiliser un canal MQI dédié pour récupérer les unités de travail incomplètes auxquelles le gestionnaire de files d'attente participait au moment de l'incident.

Notez les conditions suivantes lors de l'utilisation d'un client transactionnel étendu avec un gestionnaire de transactions compatible XA:

- Dans une même unité d'exécution, une application client ne peut être connectée qu'à un seul gestionnaire de files d'attente à la fois. Cette restriction s'applique uniquement lors de l'utilisation d'un client transactionnel étendu ; une application client qui utilise un client de base IBM MQ peut être connectée à plusieurs gestionnaires de files d'attente simultanément dans une même unité d'exécution.
- Chaque unité d'exécution d'une application client peut se connecter à un gestionnaire de files d'attente différent.
- Une application client ne peut pas utiliser de descripteurs de connexion partagée.

Pour configurer la fonction transactionnelle étendue, vous devez fournir les informations suivantes au gestionnaire de transactions pour chaque gestionnaire de files d'attente qui agit en tant que gestionnaire de ressources:

- Une chaîne xa_open
- Pointeur vers une structure de commutateur XA

Lorsque le gestionnaire de transactions appelle xa_open pour ouvrir le gestionnaire de files d'attente en tant que gestionnaire de ressources, il transmet la chaîne xa_open au client transactionnel étendu en tant qu'argument, xa_info, sur l'appel. Le client transactionnel étendu utilise les informations de la chaîne xa_open de la manière suivante:

- Pour démarrer un canal MQI vers le gestionnaire de files d'attente du serveur, si l'application client n'en a pas déjà démarré un
- Pour vérifier que le gestionnaire de files d'attente que le gestionnaire de transactions ouvre en tant que gestionnaire de ressources est identique au gestionnaire de files d'attente auquel l'application client se connecte
- Pour localiser les fonctions ax_reg et ax_unreg du gestionnaire de transactions, si le gestionnaire de files d'attente utilise l'enregistrement dynamique

Pour le format d'une chaîne xa_open et pour plus de détails sur la façon dont les informations de la chaîne xa_open sont utilisées par un client transactionnel étendu, voir [«Format d'une chaîne xa_open»](#), à la page 24.

Une structure de commutateur XA permet au gestionnaire de transactions de localiser les fonctions xa_ fournies par le client transactionnel étendu et indique si le gestionnaire de files d'attente utilise l'enregistrement dynamique. Pour plus d'informations sur les structures de commutateur XA fournies avec un client transactionnel étendu, voir [«Les structures de commutateur XA»](#), à la page 28.

Pour plus d'informations sur la configuration de la fonction transactionnelle étendue pour un gestionnaire de transactions particulier et pour toute autre information sur l'utilisation du gestionnaire de transactions avec un client transactionnel étendu, voir les sections suivantes:

- [«Configuration d'un client transactionnel étendu pour CICS»](#), à la page 29
- [«Configuration d'un client transactionnel étendu pour Tuxedo»](#), à la page 30

Concepts associés

«Paramètres CHANNEL, TRPTYPE, CONNAME et QMNAME de la chaîne xa_open», à la page 26
Utilisez ces informations pour comprendre comment le client transactionnel étendu utilise ces paramètres pour déterminer le gestionnaire de files d'attente auquel se connecter.

«Traitement d'erreurs supplémentaires pour xa_open», à la page 27
L'appel xa_open échoue dans certaines circonstances.

Tâches associées

«Utilisation du client transactionnel étendu avec des canaux TLS», à la page 28
Vous ne pouvez pas configurer un canal TLS à l'aide de la chaîne xa_open. Suivez ces instructions pour utiliser la table de définition de canal du client (ccdt).

Référence associée

«Paramètres TPM et AXLIB», à la page 27

Un client transactionnel étendu utilise les paramètres TPM et AXLIB pour localiser les fonctions ax_reg et ax_unreg du gestionnaire de transactions. Ces fonctions sont utilisées uniquement si le gestionnaire de files d'attente utilise l'enregistrement dynamique.

«Reprise à la suite d'un échec du traitement transactionnel étendu», à la page 27

A la suite d'un incident, un gestionnaire de transactions doit être en mesure de récupérer les unités d'oeuvre incomplètes. Pour ce faire, le gestionnaire de transactions doit pouvoir ouvrir en tant que gestionnaire de ressources tout gestionnaire de files d'attente participant à une unité d'oeuvre incomplète au moment de l'échec.

IBM MQ for z/OS considerations for extended transactional client connections

Some XA transaction managers use sequences of transaction coordination calls which are incompatible with the features normally available to clients connecting to IBM MQ for z/OS.

Where an incompatible sequence is detected, IBM MQ for z/OS might issue an abend for the connection and return an error response to the client.

For example, xa_prepare receives abend 5C6-00D4007D, with return code -3 (XAER_RMERR) returned to the client.

Another example is that xa_end receives abend 5C6-00D40079.

For transaction managers which encounter this situation, take the following action to allow the transaction manager to interact with IBM MQ for z/OS.

Ensure that you have enabled changes to XA client connections on IBM MQ for z/OS which allow the transaction manager to prepare a transaction on a different connection.

Notes:

- The change is not enabled by default. To make use of the change you must specify the keyword CSQSERVICE1 (in upper case) anywhere in the description field of the SVRCONN channel used by the XA client.
- Channels with the CSQSERVICE1 keyword have the following restrictions:
 - GROUP unit of recovery disposition is not permitted. Only QMGR unit of recovery disposition is allowed. The disposition is determined by the name given on the xa_open call. If the queue sharing group name is used, then the XA connection requests a group unit of recovery.

An xa_open call specifying the queue sharing group name in the **xa_info** parameter fails with *xaer_inval*.

- The `MQGMO_LOCK` and `MQGMO_UNLOCK` options are not permitted. An `MQGET` call with `MQGMO_LOCK` or `MQGMO_UNLOCK` fails with `MQRC_ENVIRONMENT_ERROR`.

The change was enabled at IBM MQ for z/OS 9.0 through [APAR P173410](#)

Related concepts

“Configuration de gestionnaires de transactions compatibles XA” on page 21

Configurez d'abord le client de base IBM MQ , puis configurez la fonction transactionnelle étendue à l'aide des informations de ces rubriques.

Format d'une chaîne xa_open

Une chaîne xa_open contient des paires de noms et de valeurs de paramètres définis.

Une chaîne xa_open a le format suivant:

```
parm_name1 = parm_value1, parm_name2 = parm_value2, ...
```

où `parm_name` est le nom d'un paramètre et `parm_value` est la valeur d'un paramètre. Les noms des paramètres ne sont pas sensibles à la casse, mais, sauf indication contraire, les valeurs des paramètres sont sensibles à la casse. Vous pouvez spécifier les paramètres dans n'importe quel ordre.

Les noms, les significations et les valeurs valides des paramètres sont les suivants:

Nom

Signification et valeurs valides

Canal

Nom d'un canal MQI.

Ce paramètre est facultatif. Si ce paramètre est indiqué, le paramètre `CONNNAME` doit également être indiqué.

TRPTYPE

Protocole de communication pour le canal MQI. Les protocoles suivants sont des valeurs valides:

LU62

LU SNA 6.2

NETBIOS

NetBIOS

SPX

IPX/SPX

TCP

TCP/IP

Ce paramètre est facultatif. S'il est omis, la valeur par défaut de TCP est prise en compte. Les valeurs du paramètre ne sont pas sensibles à la casse.

CONNNAME

Adresse réseau du gestionnaire de files d'attente à l'extrémité serveur du canal MQI. Les valeurs valides de ce paramètre dépendent de la valeur du paramètre `TRPTYPE`:

LU62

Nom de destination symbolique qui identifie une entrée d'informations côté CPI-C.

Le nom qualifié du réseau d'une LU partenaire n'est pas une valeur valide, ni un alias de LU partenaire. En effet, il n'existe pas de paramètres supplémentaires pour spécifier un nom de programme transactionnel (TP) et un nom de mode.

NETBIOS

Nom NetBIOS .

SPX

Une adresse réseau de 4 octets, une adresse de noeud de 6 octets et un numéro de socket de 2 octets facultatif. Ces valeurs doivent être spécifiées en notation hexadécimale. Un point doit

séparer les adresses réseau et de noeud, et le numéro de socket, s'il est fourni, doit être placé entre parenthèses. Exemple :

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

Si le numéro de socket est omis, la valeur par défaut 5e86 est utilisée.

TCP

Nom d'hôte ou adresse IP, éventuellement suivi d'un numéro de port entre parenthèses. Si le numéro de port est omis, la valeur par défaut 1414 est utilisée. Plusieurs hôtes et ports pour un gestionnaire de files d'attente peuvent être spécifiés à l'aide d'un séparateur de point-virgule, par exemple:

```
host1(1415);host2(1416);host3(1417)
```


Ce paramètre est facultatif. Si ce paramètre est indiqué, le paramètre CHANNEL doit également être indiqué.

QMNAME

Nom du gestionnaire de files d'attente à l'extrémité serveur du canal MQI. Le nom ne peut pas être à blanc ou un astérisque unique (*), ni commencer par un astérisque. Cela signifie que le paramètre doit identifier un gestionnaire de files d'attente spécifique par son nom.

Il s'agit d'un paramètre obligatoire.

Lorsqu'une application client est connectée à un gestionnaire de files d'attente spécifique, toute récupération de transaction doit être traitée par le même gestionnaire de files d'attente.

 Si l'application se connecte à un gestionnaire de files d'attente z/OS, elle peut spécifier le nom d'un gestionnaire de files d'attente spécifique ou le nom d'un groupe de partage de files d'attente (QSG). En utilisant le nom du gestionnaire de files d'attente ou le nom du groupe de partage de files d'attente, l'application contrôle si elle prend en compte une transaction avec une disposition d'unité de récupération QMGR ou une disposition d'unité de récupération GROUP. La disposition de l'unité de récupération GROUP permet la récupération de la transaction à traiter sur n'importe quel membre du groupe de partage de files d'attente. Pour utiliser les unités de récupération GROUP, l'attribut de gestionnaire de files d'attente **GROUPUR** doit être activé. Pour plus d'informations sur l'utilisation de l'unité de récupération GROUP, voir [Disposition de l'unité de récupération dans un groupe de partage de files d'attente](#).

TPM

Gestionnaire de transactions utilisé. Les valeurs admises sont CICS et TUXEDO.

Un client transactionnel étendu utilise ce paramètre et le paramètre AXLIB dans le même but. Pour plus d'informations sur ces paramètres, voir [Paramètres TPM et AXLIB](#).

Ce paramètre est facultatif. Les valeurs du paramètre ne sont pas sensibles à la casse.

AXLIB

Nom de la bibliothèque qui contient les fonctions ax_reg et ax_unreg du gestionnaire de transactions.

Ce paramètre est facultatif.

ID utilisateur

ID utilisateur fourni au gestionnaire de files d'attente pour l'authentification. Si ce paramètre est fourni, le paramètre **PWD** doit également être fourni. Si l'ID utilisateur et le mot de passe fournis sont authentifiés, l'ID utilisateur est utilisé pour l'identification de la connexion du gestionnaire de transactions. L'ID utilisateur et le mot de passe renseignent l'objet MQCSP sur l'appel MQCONN.

Les paramètres **UID** et **PWD** sont valides pour les liaisons client et serveur.

personnes handicapées

Mot de passe fourni au gestionnaire de files d'attente pour l'authentification. Si ce paramètre est fourni, le paramètre **UID** doit également être fourni.

Avvertissement : Dans certains cas, le mot de passe dans une structure MQCSP pour une application client est envoyé sur un réseau en texte clair. Pour vous assurer que les mots de passe d'application client sont correctement protégés, voir [Protection par mot de passe CSPIBM MQ](#).

Voici un exemple de chaîne xa_open:

```
channel=MARS.SVR, trptype=tcp, conname=MARS(1415), qmname=MARS, tpm=cics
```

Paramètres CHANNEL, TRPTYPE, CONNAME et QMNAME de la chaîne xa_open

Utilisez ces informations pour comprendre comment le client transactionnel étendu utilise ces paramètres pour déterminer le gestionnaire de files d'attente auquel se connecter.

Si les paramètres **CHANNEL** et **CONNAME** sont fournis dans la chaîne xa_open, le client transactionnel étendu utilise ces paramètres et le paramètre **TRPTYPE** pour démarrer un canal MQI vers le gestionnaire de files d'attente du serveur.

Si les paramètres **CHANNEL** et **CONNAME** ne sont pas fournis dans la chaîne xa_open, le client transactionnel étendu utilise la valeur de la variable d'environnement MQSERVER pour démarrer un canal MQI. Si la variable d'environnement MQSERVER n'est pas définie, le client transactionnel étendu utilise l'entrée de la définition de canal du client identifiée par le paramètre **QMNAME**.

Dans chacun de ces cas, le client transactionnel étendu vérifie que la valeur du paramètre **QMNAME** est le nom du gestionnaire de files d'attente à l'extrémité serveur du canal MQI. Si ce n'est pas le cas, l'appel xa_open échoue et le gestionnaire de transactions signale l'échec à l'application.

z/OS Si l'application utilise un nom de groupe de partage de files d'attente dans la zone de paramètre **QMNAME** et que la propriété GROUPUR est désactivée sur le gestionnaire de files d'attente auquel elle se connecte, l'appel xa_open échoue.

z/OS Si le client d'application se connecte à un gestionnaire de files d'attente z/OS, il peut spécifier un nom de groupe de partage de files d'attente (QSG) pour le paramètre **QMNAME**. Cela permet au client d'application de participer à une transaction avec une disposition d'unité de récupération GROUP. Pour plus d'informations sur la disposition de l'unité de récupération GROUP, voir [Disposition de l'unité de récupération](#).

Lorsque l'application client appelle ultérieurement MQCONN ou MQCONNX sur la même unité d'exécution que celle utilisée par le gestionnaire de transactions pour émettre l'appel xa_open, l'application reçoit un descripteur de connexion pour le canal MQI qui a été démarré par l'appel xa_open. Un deuxième canal MQI n'est pas démarré. Le client transactionnel étendu vérifie que la valeur du paramètre **QMgrName** dans l'appel MQCONN ou MQCONNX correspond au nom du gestionnaire de files d'attente à l'extrémité serveur du canal MQI. Si ce n'est pas le cas, l'appel MQCONN ou MQCONNX échoue avec le code anomalie MQRC_ANOTHER_Q_MGR_CONNECTED. Si la valeur du paramètre **QMgrName** est vide ou un astérisque unique (*), ou commence par un astérisque, l'appel MQCONN ou MQCONNX échoue avec le code anomalie MQRC_Q_MGR_NAME_ERROR.

Si l'application client a déjà démarré un canal MQI en appelant MQCONN ou MQCONNX avant que le gestionnaire de transactions n'appelle xa_open sur la même unité d'exécution, le gestionnaire de transactions utilise ce canal MQI à la place. Un deuxième canal MQI n'est pas démarré. Le client transactionnel étendu vérifie que la valeur du paramètre **QMNAME** dans la chaîne xa_open est le nom du gestionnaire de files d'attente du serveur. Si ce n'est pas le cas, l'appel xa_open échoue.





Si une application client démarre un canal MQI en premier, la valeur du paramètre **QMgrName** sur l'appel MQCONN ou MQCONNX peut être vide ou un astérisque (*) unique, ou elle peut commencer par un astérisque. Toutefois, dans ces circonstances, vous devez vous assurer que le gestionnaire de files d'attente auquel l'application se connecte est identique au gestionnaire de files d'attente que le gestionnaire de transactions a l'intention d'ouvrir en tant que gestionnaire de ressources lorsqu'il appelle ultérieurement xa_open sur la même unité d'exécution. Vous risquez de rencontrer moins de problèmes, par conséquent, si la valeur du paramètre **QMgrName** identifie explicitement le gestionnaire de files d'attente par son nom.

Paramètres TPM et AXLIB

Un client transactionnel étendu utilise les paramètres TPM et AXLIB pour localiser les fonctions ax_reg et ax_unreg du gestionnaire de transactions. Ces fonctions sont utilisées uniquement si le gestionnaire de files d'attente utilise l'enregistrement dynamique.

Si le paramètre TPM est fourni dans une chaîne xa_open, mais que le paramètre AXLIB n'est pas fourni, le client transactionnel étendu prend une valeur pour le paramètre AXLIB en fonction de la valeur du paramètre TPM. Voir [Tableau 3](#), à la page 27 pour les valeurs supposées du paramètre AXLIB.

Tableau 3. Valeurs par défaut du paramètre AXLIB

Valeur de TPM	Plateforme	Valeur par défaut d'AXLIB
CICS	 AIX	/usr/lpp/encina/lib/libEncServer.a(EncServer_shr.o)
CICS	Systèmes  Windows	Serveur libEnc
Tuxedo	 AIX	/usr/lpp/tuxedo/lib/libtux.a(libtux.so.60)
Tuxedo	Systèmes  Windows	libtux

Si le paramètre AXLIB est fourni dans une chaîne xa_open, le client transactionnel étendu utilise sa valeur pour remplacer toute valeur supposée basée sur la valeur du paramètre TPM. Le paramètre AXLIB peut également être utilisé pour un gestionnaire de transactions pour lequel le paramètre TPM n'a pas de valeur spécifiée.

Traitement d'erreurs supplémentaires pour xa_open

L'appel xa_open échoue dans certaines circonstances.

Les rubriques de cette section décrivent les situations dans lesquelles l'appel xa_open échoue. Elle échoue également si l'une des situations suivantes se produit:

- Il y a des erreurs dans la chaîne xa_open.
- Les informations sont insuffisantes pour démarrer un canal MQI.
- Un problème s'est produit lors de la tentative de démarrage d'un canal MQI (le gestionnaire de files d'attente du serveur n'est pas en cours d'exécution, par exemple).

Reprise à la suite d'un échec du traitement transactionnel étendu

A la suite d'un incident, un gestionnaire de transactions doit être en mesure de récupérer les unités d'oeuvre incomplètes. Pour ce faire, le gestionnaire de transactions doit pouvoir ouvrir en tant que gestionnaire de ressources tout gestionnaire de files d'attente participant à une unité d'oeuvre incomplète au moment de l'échec.

Par conséquent, vous devez vous assurer que toutes les unités de travail incomplètes ont été résolues avant d'apporter des modifications aux informations de configuration.

Vous devez également vous assurer que les modifications apportées à la configuration n'affectent pas la capacité du gestionnaire de transactions à ouvrir les gestionnaires de files d'attente qu'il doit ouvrir. Voici des exemples de ces changements de configuration:

- Modification du contenu d'une chaîne xa_open
- Modification de la valeur de la variable d'environnement MQSERVER
- Modification des entrées de la table de définition de canal du client (CCDT)
- Suppression d'une définition de canal de connexion serveur

Les structures de commutateur XA

Deux structures de commutateur XA sont fournies avec le client transactionnel étendu sur chaque plateforme.

Ces structures de commutateur sont les suivantes:




MQRMIXASwitch

Cette structure de commutation est utilisée par un gestionnaire de transactions lorsqu'un gestionnaire de files d'attente, agissant en tant que gestionnaire de ressources, n'utilise pas l'enregistrement dynamique.

MQRMIXASwitchDynamic

Cette structure de commutation est utilisée par un gestionnaire de transactions lorsqu'un gestionnaire de files d'attente, agissant en tant que gestionnaire de ressources, utilise l'enregistrement dynamique.

Ces structures de commutateur se trouvent dans les bibliothèques indiquées dans la [Tableau 4](#), à la page 28.

Plateforme	Bibliothèque contenant les structures de commutateur XA
 AIX	MQ_INSTALLATION_PATH/lib/libmqcxa
 Linux	
Systèmes  Windows	MQ_INSTALLATION_PATH\bin\mqcxa.dll ¹

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

Le nom du gestionnaire de ressources IBM MQ dans chaque structure de commutateur est MQSeries_XA_RMI, mais de nombreux gestionnaires de files d'attente peuvent partager la même structure de commutateur.

Concepts associés

«Enregistrement dynamique et traitement transactionnel étendu», à la page 28

L'utilisation de l'enregistrement dynamique est une forme d'optimisation car elle peut réduire le nombre d'appels xa _ fonction émis par le gestionnaire de transactions.

Enregistrement dynamique et traitement transactionnel étendu

L'utilisation de l'enregistrement dynamique est une forme d'optimisation car elle peut réduire le nombre d'appels xa _ fonction émis par le gestionnaire de transactions.

Si un gestionnaire de files d'attente n'utilise pas l'enregistrement dynamique, un gestionnaire de transactions implique le gestionnaire de files d'attente dans chaque unité d'oeuvre. Le gestionnaire de transactions effectue cette opération en appelant xa_start, xa_end et xa_prepare, même si le gestionnaire de files d'attente ne dispose d'aucune ressource mise à jour dans l'unité d'oeuvre.

Si un gestionnaire de files d'attente utilise l'enregistrement dynamique, un gestionnaire de transactions démarre en supposant que le gestionnaire de files d'attente n'est pas impliqué dans une unité d'oeuvre et n'appelle pas xa_start. Le gestionnaire de files d'attente est alors impliqué dans l'unité d'oeuvre uniquement si ses ressources sont mises à jour dans le cadre du contrôle de point de synchronisation. Dans ce cas, le client transactionnel étendu appelle ax_reg pour enregistrer l'implication du gestionnaire de files d'attente.

Utilisation du client transactionnel étendu avec des canaux TLS

Vous ne pouvez pas configurer un canal TLS à l'aide de la chaîne xa_open. Suivez ces instructions pour utiliser la table de définition de canal du client (ccdt).

Pourquoi et quand exécuter cette tâche

En raison de la taille limitée de la chaîne xa_open xa_info, il n'est pas possible de transmettre toutes les informations requises pour configurer un canal TLS à l'aide de la méthode de chaîne xa_open de connexion à un gestionnaire de files d'attente. Par conséquent, vous devez utiliser la table de définition de canal du client ou, si votre gestionnaire de transactions l'autorise, créer le canal avec MQCONNX avant d'émettre l'appel xa_open.

Pour utiliser la table de définition de canal du client, procédez comme suit:

Procédure

1. Spécifiez une chaîne xa_open contenant uniquement le paramètre obligatoire qmname (nom du gestionnaire de files d'attente), par exemple: XA_Open_String=qmname=MYQM
2. Utilisez un gestionnaire de files d'attente pour définir un canal CLNTCONN (connexion client) avec les paramètres TLS requis. Incluez le nom du gestionnaire de files d'attente dans l'attribut QMNAME de la définition CLNTCONN. Il sera mis en correspondance avec le qmname dans la chaîne xa_open.
3. Mettez la définition CLNTCONN à la disposition du système client dans une table de définition de canal du client (CCDT) ou, sous Windows, dans le répertoire actif.
4. Si vous utilisez une table CCDT, identifiez la table CCDT contenant la définition du canal CLNTCONN à l'aide des variables d'environnement MQCHLLIB et MQCHLTAB. Définissez ces variables dans les environnements utilisés par l'application client et le gestionnaire de transactions.

Résultats

Cela donne au gestionnaire de transactions une définition de canal pour le gestionnaire de files d'attente approprié avec les attributs TLS nécessaires pour s'authentifier correctement, y compris SSLCIPH, CipherSpec.

Configuration d'un client transactionnel étendu pour CICS

Vous configurez un client transactionnel étendu à utiliser par CICS en ajoutant une définition de ressource XAD à une région CICS .

Ajoutez la définition de ressource XAD à l'aide de la commande CICS resource definition online (RDO), **cicsadd**. La définition de ressource XAD spécifie les informations suivantes:

- Une chaîne xa_open
- Nom de chemin qualifié complet d'un fichier de commutation de chargement

Un fichier de commutation de chargement est fourni pour être utilisé par CICS sur chacune des plateformes suivantes:

-  AIX
-  Windows

Chaque fichier de chargement de commutateur contient une fonction qui renvoie un pointeur vers la structure de commutateur XA utilisée pour l'enregistrement dynamique, MQRMIXASwitchDynamic. Voir [Tableau 5](#), à la [page 29](#) pour le nom de chemin qualifié complet de chaque fichier de commutation de chargement.



Plateforme	Fichier de commutation de chargement
 AIX	MQ_INSTALLATION_PATH/lib/amqczsc
 Linux	

Tableau 5. Fichiers de commutation de chargement (suite)	
Plateforme	Fichier de commutation de chargement
Windows	MQ_INSTALLATION_PATH\bin\mqcc4swi.dll ¹

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

Voici un exemple de définition de ressource XAD pour les systèmes Windows :

```
cicsadd -c xad -r REGION1 WMQXA \
  ResourceDescription="IBM MQ queue manager MARS" \
  XAOpen="channel=MARS.SVR, trptype=tcp, conname=MARS(1415), qmname=MARS, tpm=cics" \
  SwitchLoadFile="C:\Program Files\IBM\MQ\bin\mqcc4swi.dll"
```

Pour plus d'informations sur l'ajout d'une définition de ressource XAD à une région CICS , voir *CICS Administration Reference* et *CICS Administration Guide* pour votre plateforme.

Notez les informations suivantes sur l'utilisation de CICS avec un client transactionnel étendu:

- Vous ne pouvez ajouter qu'une seule définition de ressource XAD pour IBM MQ à une région CICS . Cela signifie qu'un seul gestionnaire de files d'attente peut être associé à une région et que toutes les applications CICS qui s'exécutent dans la région ne peuvent se connecter qu'à ce gestionnaire de files d'attente. Si vous souhaitez exécuter des applications CICS qui se connectent à un gestionnaire de files d'attente différent, vous devez exécuter les applications dans une région différente.
- Chaque serveur d'applications d'une région appelle xa_open lors de son initialisation et démarre un canal MQI vers le gestionnaire de files d'attente associé à la région. Cela signifie que le gestionnaire de files d'attente doit être démarré avant le démarrage d'un serveur d'applications, sinon l'appel xa_open échoue. Toutes les applications IBM MQ MQI client traitées ultérieurement par le serveur d'applications utilisent le même canal MQI.
- Lorsqu'un canal MQI démarre et qu'il n'y a pas d'exit de sécurité à l'extrémité client du canal, l'ID utilisateur qui transite du système client vers l'agent MCA de connexion serveur est cics . Dans certaines circonstances, le gestionnaire de files d'attente utilise cet ID utilisateur pour vérifier les droits d'accès lorsque l'agent MCA de connexion serveur tente ensuite d'accéder aux ressources du gestionnaire de files d'attente pour le compte d'une application client. Si cet ID utilisateur est utilisé pour les vérifications des droits d'accès, vous devez vous assurer qu'il dispose des droits d'accès à toutes les ressources auxquelles il a besoin d'accéder.

Pour plus d'informations sur le moment où le gestionnaire de files d'attente utilise cet ID utilisateur pour les vérifications des droits d'accès, voir [Sécurisation](#).

- Les exits d'arrêt de tâche CICS qui sont fournis pour être utilisés sur les systèmes client IBM MQ sont répertoriés dans la rubrique [Tableau 6, à la page 30](#) . Vous configurez ces exits de la même manière que vous configurez les exits correspondants pour les systèmes serveur IBM MQ . Pour plus d'informations, voir [Activation des exits utilisateur CICS](#).

Tableau 6. exit de fin de tâche CICS		
Plateforme	Source	Bibliothèque
AIX	amqzscgx.c	amqzscg
Linux		
Systèmes Windows	amqzscgn.c	mqcc1415.dll

Configuration d'un client transactionnel étendu pour Tuxedo

Pour configurer la définition de ressource XAD à utiliser par Tuxedo, mettez à jour le fichier UBBCONFIG et la table du gestionnaire de ressources.

Pour configurer la définition de ressource XAD à utiliser par Tuxedo, procédez comme suit:

- Dans la section GROUPS du fichier Tuxedo UBBCONFIG pour une application, utilisez le paramètre **OPENINFO** pour spécifier une chaîne xa_open. Pour obtenir un exemple de cette procédure, consultez l'exemple de fichier UBBCONFIG, qui est fourni pour être utilisé avec les exemples de programmes Tuxedo.

AIX Sur les plateformes suivantes, le nom du fichier est ubbstxcx.cfg:

– AIX

Windows Windows, le nom du fichier est ubbstxcn.cfg.

- Dans l'entrée d'un gestionnaire de files d'attente de la table de gestionnaires de ressources Tuxedo, indiquez le nom d'une structure de commutateur XA et le nom de chemin qualifié complet de la bibliothèque qui contient la structure:

– **AIX** Sous AIX, spécifiez udataobj/RM.

– **Windows** Sous Windows, spécifiez udataobj\rm.

Pour obtenir un exemple de cette procédure pour chaque plateforme, voir [Exemples TUXEDO](#). Tuxedo prend en charge l'enregistrement dynamique d'un gestionnaire de ressources et vous pouvez donc utiliser MQRMIXASwitch ou MQRMIXASwitchDynamic.

Windows Serveur de transactions Microsoft

Aucune configuration supplémentaire n'est requise pour que vous puissiez utiliser Microsoft Transaction Server (MTS) en tant que gestionnaire de transactions. Cependant, il y a quelques points à noter.

Notez les informations suivantes sur l'utilisation de MTS avec le client transactionnel étendu:

- Une application MTS démarre toujours un canal MQI lorsqu'elle se connecte à un gestionnaire de files d'attente de serveur. MTS, dans son rôle de gestionnaire de transactions, utilise ensuite le même canal MQI pour communiquer avec le gestionnaire de files d'attente.
- Suite à une défaillance, MTS doit être en mesure de récupérer toutes les unités de travail incomplètes. Pour ce faire, MTS doit être en mesure de communiquer avec tout gestionnaire de files d'attente qui participait à une unité de travail incomplète au moment de l'échec.

Lorsqu'une application MTS se connecte à un gestionnaire de files d'attente de serveur et démarre un canal MQI, le client transactionnel étendu extrait suffisamment d'informations des paramètres de l'appel MQCONN ou MQCONNX pour permettre le redémarrage du canal suite à un échec, si nécessaire. Le client transactionnel étendu transmet les informations à MTS, qui les enregistre dans son journal.

Si l'application MTS émet un appel MQCONN, ces informations sont simplement le nom du gestionnaire de files d'attente. Si l'application MTS émet un appel MQCONNX et fournit une structure de définition de canal, MQCD, les informations incluent également le nom du canal MQI, l'adresse réseau du gestionnaire de files d'attente du serveur et le protocole de communication du canal.

Dans une situation de reprise, MTS renvoie ces informations au client transactionnel étendu et ce dernier les utilise pour redémarrer le canal MQI.

Si vous devez modifier des informations de configuration, assurez-vous que toutes les unités d'œuvre incomplètes ont été résolues avant d'effectuer les modifications. Vous pouvez également vous assurer que les modifications de configuration n'affectent pas la capacité du client transactionnel étendu à redémarrer un canal MQI à l'aide des informations enregistrées par MTS. Voici des exemples de ces changements de configuration:

- Modification de la valeur de la variable d'environnement MQSERVER
 - Modification des entrées de la table de définition de canal du client (CCDT)
 - Suppression d'une définition de canal de connexion serveur
- Notez les conditions suivantes lors de l'utilisation d'un client transactionnel étendu avec MTS:

- Dans une même unité d'exécution, une application client ne peut être connectée qu'à un seul gestionnaire de files d'attente à la fois.
- Chaque unité d'exécution d'une application client peut se connecter à un gestionnaire de files d'attente différent.
- Une application client ne peut pas utiliser de descripteurs de connexion partagée.

Définition de canaux MQI

Pour créer un canal, vous devez créer **deux** définitions de canal, une pour chaque extrémité de la connexion, en utilisant le même nom de canal et des types de canal compatibles. Dans ce cas, les types de canal sont *server-connection* et *client-connection*.

Canaux définis par l'utilisateur

Lorsque le serveur ne définit pas automatiquement les canaux, il existe deux façons de créer les définitions de canal et de donner à l'application IBM MQ sur la machine IBM MQ MQI client l'accès au canal.

Ces deux méthodes sont décrites en détail:

1. Créez une définition de canal sur le client IBM MQ et l'autre sur le serveur.

Cela s'applique à toute combinaison de IBM MQ MQI client et de plateformes de serveur. Utilisez-le lorsque vous démarrez sur le système ou pour tester votre configuration.

Pour plus de détails sur l'utilisation de cette méthode, voir [«Création de définitions de connexion serveur et de connexion client sur différentes plateformes»](#), à la page 38 .

2. Créez les deux définitions de canal sur la machine serveur.

Utilisez cette méthode lorsque vous configurez simultanément plusieurs canaux et des machines IBM MQ MQI client .

Pour plus de détails sur l'utilisation de cette méthode, voir [«Création de définitions de connexion serveur et de connexion client sur le serveur»](#), à la page 44 .

Canaux définis automatiquement

Multi

Les produits IBM MQ sur Multiplatforms incluent une fonction qui permet de créer automatiquement une définition de canal sur le serveur s'il n'en existe pas.

Si une demande d'association entrante est reçue d'un client et qu'une définition de connexion serveur appropriée est introuvable sur ce gestionnaire de files d'attente, IBM MQ crée automatiquement une définition et l'ajoute au gestionnaire de files d'attente. La définition automatique est basée sur la définition du canal de connexion serveur par défaut SYSTEM.AUTO.SVRCONN. Vous activez la définition automatique des définitions de connexion serveur en mettant à jour l'objet gestionnaire de files d'attente à l'aide de la commande ALTER QMGR avec le paramètre CHAD (ou de la commande PCF Change Queue Manager avec le paramètre ChannelAutoDef).

Concepts associés

[«Fonction de contrôle de canal»](#), à la page 244

La fonction de contrôle de canal fournit des fonctions permettant de définir, de surveiller et de contrôler des canaux.





ALW

Création et utilisation de canaux AMQP

Lorsque vous installez le support IBM MQ pour le composant de service AMQP dans votre installation IBM MQ , vous pouvez exécuter des commandes IBM MQ MQSC (**runmqsc**) pour définir, modifier, supprimer, démarrer et arrêter un canal. Vous pouvez aussi afficher le statut d'un canal.

Avant de commencer

Cette tâche suppose que vous avez installé le canal AMQP. Pour ce faire, sélectionnez le composant Service AMQP lorsque vous installez IBM MQ. Pour plus d'informations, suivez le lien correspondant à votre plateforme, puis recherchez la ligne de table "AMQP Service":

-  [Composants IBM MQ pour les systèmes AIX](#)
-  [IBM MQ pour les systèmes Linux](#)
-  [IBM MQ Debian pour les systèmes Linux Ubuntu](#)
-  [Fonctions d'IBM MQ pour les systèmes Windows](#)

Remarque : Voir [Redémarrage du service IBM MQ pour AMQP](#) pour un exemple de composant SERVICE et plus d'informations si votre service AMQP cesse de fonctionner correctement.

Cette tâche suppose également que vous disposez d'un gestionnaire de files d'attente existant.

Pour établir une connexion de test au gestionnaire de files d'attente, vous pouvez utiliser tous les clients AMQP qui implémentent le protocole OASIS AMQP 1.0 , tels que les clients MQ Light et Apache Qpid tels que Apache Qpid Proton et Apache Qpid JMS.

Depuis la IBM MQ 9.3.0, vous ne pouvez utiliser que le canal par défaut, SYSTEM.DEF.AMQP, pour tester les connexions MQ Light au gestionnaire de files d'attente. La procédure ci-après utilise le canal par défaut.



Cette tâche repose sur le client MQ Light Node.js. Toutefois, les étapes relatives au gestionnaire de files d'attente IBM MQ sont valables pour tous les clients.

Remarque : Les canaux AMQP ne prennent pas en charge les services AMQP définis par l'utilisateur. Les canaux AMQP ne prennent en charge que le service SYSTEM.AMQP.SERVICE par défaut du système. Vous ne pouvez définir qu'une seule instance de ce service pour chaque gestionnaire de files d'attente.

Procédure



1. Démarrez **runmqsc** à partir du répertoire `mqinstall/bin/` :

```
runmqsc QMNAME
```

2. (Nécessaire uniquement si votre gestionnaire de files d'attente est   de IBM MQ 9.4.0 ou s'il s'agit de IBM MQ 9.0.4 ou d'une version antérieure.) Assurez-vous que la fonction AMQP est installée et fonctionne correctement.

Utilisez la commande **START SERVICE** pour démarrer le service IBM MQ , qui contrôle la machine virtuelle Java:

```
START SERVICE(SYSTEM.AMQP.SERVICE)
```

Remarque :   Depuis la IBM MQ 9.4.0 , l'attribut **CONTROL** de SYSTEM.AMQP.SERVICE est défini sur MANUEL. Cela empêche le service d'être démarré lorsque le gestionnaire de files d'attente est démarré. La définition de la propriété **CONTROL** sur QMGRdémarré automatiquement lorsque le gestionnaire de files d'attente est démarré.

De IBM MQ 9.1 à IBM MQ 9.3, SYSTEM.AMQP.SERVICE a son attribut **CONTROL** défini sur QMGR.

3. Définissez l'ID utilisateur MCAUSER.

Lorsqu'un client AMQP se connecte à un canal, le canal spécifie un ID utilisateur MCAUSER, qui est utilisé pour les connexions au gestionnaire de files d'attente. La valeur par défaut de MCAUSER est vide. Pour que les clients AMQP puissent se connecter au gestionnaire de files d'attente, vous devez spécifier une valeur MCAUSER , qui doit être un utilisateur IBM MQ valide autorisé à publier et à s'abonner aux rubriques IBM MQ .

a) Utilisez la commande **ALTER CHANNEL** pour définir l'ID utilisateur MCAUSER :

```
ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) MCAUSER(User ID)
```

b) Utilisez les deux commandes **setmqaut** suivantes pour autoriser votre ID utilisateur MCAUSER à publier des informations et à s'abonner à des rubriques:

```
setmqaut -m QMNAME -t topic -n SYSTEM.BASE.TOPIC -p MCAUSER  
-all +pub +sub
```

et

```
setmqaut -m QMNAME -t qmgr -p MCAUSER -all +connect
```

Si le canal s'exécute alors que l'ID utilisateur MCAUSER est ajouté ou altéré, vous devez l'arrêter et le redémarrer.

Remarque : Si l'ID utilisateur MCAUSER n'est pas défini ou que l'ID utilisateur MCAUSER n'est pas autorisé à publier des rubriques IBM MQ ou à s'y abonner, vous recevrez un message d'erreur dans le client AMQP.

4. Utilisez la commande **START CHANNEL** pour démarrer le système SYSTEM.DEF.AMQP AMQP:

```
START CHANNEL(SYSTEM.DEF.AMQP)
```

5. Si vous souhaitez vérifier le statut du canal, utilisez la commande **DISPLAY CHSTATUS** :

```
DISPLAY CHSTATUS(SYSTEM.DEF.AMQP) CHLTYPE(AMQP)
```

Lorsque le canal s'exécute correctement, STATUS (RUNNING) est affiché dans le résultat de la commande.

6. Changez le port par défaut.

Le port par défaut des connexions AMQP 1.0 est 5672. Si vous utilisez déjà le port 5672, ce qui est possible si vous avez installé MQ Light précédemment, vous devez changer le port que votre canal AMQP utilise. Utilisez la commande **ALTER CHANNEL** pour modifier le port:

```
ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) PORT(NEW PORT NUMBER)
```

7. Si vous ne voulez pas bloquer ou filtrer les connexions au canal AMQP avec des règles d'authentification de canal (CHLAUTH), désactivez l'authentification de canal dans le gestionnaire de files d'attente comme suit :

```
alter qmgr chlauth(disabled)
```

Il n'est pas recommandé de désactiver l'authentification de connexion dans un gestionnaire de files d'attente de production. Ne la désactivez que dans un environnement de développement.

Vous pouvez aussi configurer les règles d'authentification de canal du gestionnaire de files d'attente pour permettre des connexions spécifiques au canal AMQP.

8. Facultatif : Si vous voulez activer le chiffrement SSL/TLS sur le canal, à l'aide du référentiel de clés pour le gestionnaire de files d'attente, vous devez associer l'attribut SSLCIPH pour le canal à une spécification de chiffrement appropriée. Par défaut, la spécification de chiffrement est vide, ce qui signifie que le chiffrement SSL/TLS n'est pas utilisé sur le canal. Utilisez la commande **ALTER CHANNEL** pour définir une spécification de chiffrement. Exemple :

```
ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) SSLCIPH(CIPHER SPECIFICATION)
```

De plus, plusieurs autres options de configuration de canal sont associées au chiffrement SSL/TLS, que vous pouvez définir comme suit :

- Par défaut, 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 le nom utilisé par le chiffrement SSL/TLS pour le canal. Vous pouvez sélectionner un autre certificat en définissant **CERTLABL**. Utilisez la commande **ALTER CHANNEL** pour spécifier le libellé du certificat requis:

```
ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) CERTLABL(CERTIFICATE LABEL)
```

- Vous pouvez configurer le canal afin de demander un certificat provenant des connexions client SSL/TLS. Vous pouvez indiquer si un certificat est requis à partir d'une connexion client SSL/TLS en définissant l'attribut **SSLCAUTH**. La commande **ALTER CHANNEL** permet de définir si un certificat est requis à partir d'une connexion client SSL/TLS. Exemple :

```
ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) SSLCAUTH(REQUIRED or OPTIONAL)
```

- Si vous définissez l'attribut **SSLCAUTH** sur **REQUIRED**, le nom distinctif (DN) du certificat du client peut être vérifié. Pour vérifier le nom distinctif du certificat du client, définissez l'attribut **SSLPEER**. Utilisez la commande **ALTER CHANNEL** pour vérifier le nom distinctif du certificat du client. Exemple :

```
ALTER CHANNEL(SYSTEM.DEF.AMQP) CHLTYPE(AMQP) SSLPEER (DN SPECIFICATION)
```

Vous pouvez également utiliser des enregistrements d'authentification de canal pour autoriser ou bloquer les connexions car cette méthode offre une granularité plus élevée que l'utilisation de l'attribut **SSLPEER**. Pour plus d'informations sur la définition de **SSLPEER** et l'utilisation des enregistrements d'authentification de canal comme alternative, voir [Homologue SSL](#).

9. Installez le client MQ Light Node.js en exécutant la commande suivante :

```
npm install mqlight
```

10. Accédez au répertoire `node_modules/mqlight/samples` et exécutez l'exemple d'application de réception :

- Si vous utilisez le numéro de port par défaut, vous pouvez exécuter l'exemple d'application de réception :

```
node recv.js
```

- Si vous avez configuré votre canal AMQP pour qu'il utilise un numéro de port différent, vous pouvez exécuter l'exemple d'application de réception avec un paramètre afin de spécifier le nouveau numéro de port :

```
node recv.js -s amqp://localhost:6789
```

Si la connexion au canal par défaut aboutit, le message suivant s'affiche :

```
Connected to amqp://localhost:5672 using client-id recv_e79c55d
Subscribed to pattern: public
```

L'application est maintenant connectée au gestionnaire de files d'attente et attend de recevoir des messages. Elle est abonnée à la rubrique `public`.

Remarque : `client-id` est automatiquement généré, sauf si vous en spécifiez un à l'aide du paramètre `-i`.



11. Dans une nouvelle fenêtre de commande, accédez au répertoire `node_modules/mqlight/samples` et exécutez l'exemple d'application d'envoi en émettant la commande suivante :

```
node send.js
```

Dans la fenêtre de commande de l'application de réception, le message Hello World s'affiche.

12. Utilisez l'exemple **AMQSSUB** IBM MQ pour recevoir un exemple de message MQ Light .

Sous Linux et Windows, l'exemple se trouve dans les emplacements suivants :

-  Répertoire `mqinstall/samp/bin` sur Linux.
-  Répertoire `mqinstall/Tools\c\Samples\Bin` sur Windows.

a) Exécutez l'exemple en émettant la commande suivante :

```
amqssub public QM-name.
```

b) Envoyez un message à l'application IBM MQ en réexécutant la commande suivante :

```
node send.js
```

13. Utilisez la commande **DEFINE CHANNEL** pour créer d'autres canaux AMQP:

```
DEFINE CHANNEL(MY.AMQP.CHANNEL) CHLTYPE(AMQP) PORT(2345)
```

Lorsque vous définissez un canal, celui-ci doit être démarré manuellement avec la commande **START CHANNEL** :

```
START CHANNEL(MY.AMQP.CHANNEL)
```

Pour vérifier que le canal s'exécute correctement, vous pouvez exécuter l'exemple d'application de réception en spécifiant le port du nouveau canal :

```
node recv.js -s amqp://localhost:2345
```

Que faire ensuite

Vous pouvez utiliser les commandes suivantes pour afficher les connexions IBM MQ, arrêter le canal et supprimer le canal :

DISPLAY CONN(*) TYPE(CONN) WHERE (CHANNEL EQ SYSTEM.DEF.AMQP)

Affiche la connexion IBM MQ que le canal AMQP a établie avec le gestionnaire de files d'attente.

DISPLAY CHSTATUS(*) CHLTYPE(AMQP) CLIENTID(*) ALL

Affiche la liste des clients AMQP connectés au canal spécifié.

STOP CHANNEL (MY.AMQP.CHANNEL)

Arrête un canal AMQP et ferme le port sur lequel il est à l'écoute.

DELETE CHANNEL (MY.AMQP.CHANNEL)

Permet de supprimer un canal que vous avez créé.

Remarque : Ne supprimez pas le canal par défaut SYSTEM.DEF.AMQP.

Vous pouvez déterminer si la capacité AMQP est installée dans l'installation IBM MQ et si un gestionnaire de files d'attente est associé avec la commande **runmqsc** ou PCF :

- A l'aide de **runmqsc**, affichez les attributs du gestionnaire de files d'attente et recherchez AMQPCAP (YES).
- A l'aide de PCF, utilisez la commande **MQCMD_INQUIRE_Q_MGR** et confirmez la valeur de MQIA_AMQP_CAPABILITY.

Tâches associées

[Développement d'applications client AMQP](#)

[Sécurisation des clients AMQP](#)

Référence associée

[strmqm](#)

ALW Suppression du canal AMQP des gestionnaires de files d'attente

Vous pouvez supprimer le canal AMQP des gestionnaires de files d'attente en supprimant les dossiers dans le répertoire d'installation.

Procédure

1. Arrêtez le gestionnaire de files d'attente.
2. Retirez la prise en charge de IBM MQ pour les API de composant de service AMQP:

- **AIX** Sous AIX, exécutez la commande suivante :

```
installp -u mqm.amqp.rte
```

- **Linux** Sous Linux, supprimez le RPM AMQP. Si vous avez refait le package avant de l'installer, spécifiez le nom du package RPM refait.

```
rpm -e MQSeriesAMQP
```

- **Windows** Sous Windows, supprimez le dossier amqp de l'installation IBM MQ . Veillez à ne pas supprimer d'autres fichiers ou dossiers du chemin d'installation d'IBM MQ.

3. Redémarrez le gestionnaire de files d'attente.

Tâches associées

[Développement d'applications client AMQP](#)

[Sécurisation des clients AMQP](#)

ALW Fichiers journaux des canaux AMQP

Les fichiers journaux pour les canaux AMQP sont stockés dans le même répertoire de données IBM MQ que les fichiers journaux d'IBM MQ.

Le répertoire de données par défaut sous Windows est C:\ProgramData\IBM\MQ.

Le répertoire de données par défaut sous Linux est /var/mqm.

Le canal AMQP écrit des informations de journal dans les fichiers journaux suivants, qui se trouvent dans le répertoire de données d'IBM MQ :

- amqp.stdout, écrit dans le dossier qmgrs/QM-name .
- amqp.stderr, écrit dans le dossier qmgrs/QM-name .
- amqp_*.log , écrit dans le dossier qmgrs/QM-name/errors .
- **V9.4.0** **V9.4.0** amqp_*.json , écrit dans le dossier qmgrs/QM-name/errors .

Si un client MQ Light reçoit une erreur d'authentification ou d'autorisation, votre administrateur peut trouver des informations détaillées sur la raison de l'échec de la sécurité dans le fichier amqp_0.log ,

V9.4.0 **V9.4.0** amqp_0.json fileet les fichiers MQ AMQERR*.log .

Tous les fichiers FDC sont créés en tant que fichiers AMQP*.FDC , qui sont écrits dans le dossier *data-directory/errors* .

Certains fichiers de configuration sont écrits dans le répertoire `qmgrs/QM-name/amqp`.

V 9.4.0 **V 9.4.0** Les journaux au format JSON dans AMQP sont facultatifs et doivent être activés manuellement. Pour ce faire, modifiez les [journaux AMQP](#), les [journaux d'erreurs](#) et les [fichiers de configuration](#).

Concepts associés

[Journaux d'erreurs sur AIX, Linux, and Windows](#)

Tâches associées

[Développement d'applications client AMQP](#)

[Sécurisation des clients AMQP](#)

[Activation des journaux au format JSON pour AMQP](#)

Création de définitions de connexion serveur et de connexion client sur différentes plateformes

Vous pouvez créer chaque définition de canal sur l'ordinateur auquel elle s'applique. Toutefois, il existe des restrictions sur la façon dont vous pouvez créer des définitions de canal sur un ordinateur client.

Pourquoi et quand exécuter cette tâche

Sur toutes les plateformes, vous pouvez utiliser des commandes IBM MQ Script (MQSC), des commandes PCF (Programmable Command Format) ou IBM MQ Explorer pour définir un canal de connexion serveur sur la machine serveur.

z/OS Sous z/OS, vous pouvez également utiliser les panneaux Opération et Contrôle.

IBM i Sous IBM i, vous pouvez également utiliser l'interface du panneau.

Etant donné que les commandes MQSC ne sont pas disponibles sur une machine sur laquelle IBM MQ a été installé en tant que IBM MQ MQI client uniquement, vous devez utiliser différentes méthodes pour définir un canal de connexion client sur la machine client.

Les remarques suivantes s'appliquent lorsque **runmqsc**:

- Vous pouvez spécifier le paramètre **-c** et, éventuellement, le paramètre **-u** pour connecter **runmqsc** en tant que client au gestionnaire de files d'attente que vous souhaitez administrer.
- Si vous utilisez le paramètre **-u** pour fournir un ID utilisateur, vous êtes invité à indiquer un mot de passe correspondant.
- Si vous avez configuré l'enregistrement CONNAUTH AUTHINFO avec CHCKLOCL (REQUIRED) ou CHCKLOCL (REQDADM), vous devez utiliser le paramètre **-u**, sinon vous ne pourrez pas administrer votre gestionnaire de files d'attente avec **runmqsc**.

Procédure

- Pour définir un canal de connexion serveur sur le serveur, voir [«Définition d'un canal de connexion serveur sur le serveur»](#), à la page 38.
- Pour créer un canal de connexion client sur un IBM MQ MQI client à l'aide de la variable d'environnement **MQSERVER**, voir [«Création d'un canal de connexion client sur le IBM MQ MQI client à l'aide de MQSERVER»](#), à la page 39.
- Pour créer un canal de connexion client sur un IBM MQ MQI client à l'aide de la structure MQCNO sur un appel MQCONN, voir [«Création d'un canal de connexion client sur le IBM MQ MQI client à l'aide de MQCNO»](#), à la page 43.

Définition d'un canal de connexion serveur sur le serveur

Démarrez MQSC si nécessaire, puis définissez le canal de connexion serveur.

Procédure

1. Facultatif : Si vous utilisez un serveur Multiplatforms, créez et démarrez d'abord un gestionnaire de files d'attente, puis lancez les commandes MQSC.

- a) Créez un gestionnaire de files d'attente appelé QM1 , par exemple:

```
crtmqm QM1
```

- b) Démarrez le gestionnaire de files d'attente :

```
strmqm QM1
```

- c) Démarrez les commandes MQSC:

```
runmqsc QM1
```

2. Définissez un canal avec le nom de votre choix et le type de canal *server-connection*.

```
DEFINE CHANNEL(CHAN1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +  
DESCR('Server-connection to Client_1')
```

Cette définition de canal est associée au gestionnaire de files d'attente en cours d'exécution sur le serveur.

3. Utilisez la commande suivante pour autoriser l'accès de connexion entrante à votre gestionnaire de files d'attente:

```
SET CHLAUTH(CHAN1) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- où SET CHLAUTH utilise le nom du canal défini à l'étape précédente.
- Où 'Adresse IP' est l'adresse IP du client.
- Où 'userid' est l'ID que vous souhaitez fournir au canal pour le contrôle d'accès aux files d'attente cible. Cette zone est sensible à la casse.

Vous pouvez choisir d'identifier votre connexion entrante à l'aide d'un certain nombre d'attributs différents. L'exemple utilise l'adresse IP. Les autres attributs incluent l'ID utilisateur du client et le nom distinctif du sujet TLS. Pour plus d'informations, voir [Enregistrements d'authentification de canal](#)

Création d'un canal de connexion client sur le IBM MQ MQI client à l'aide de MQSERVER

Vous pouvez définir un canal de connexion client sur un poste de travail client à l'aide de la variable d'environnement **MQSERVER** .

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser la variable d'environnement **MQSERVER** pour spécifier une définition simple d'un canal de connexion client. Il est simple dans le sens où vous ne pouvez spécifier que quelques attributs du canal à l'aide de cette méthode.

Si vous utilisez la variable d'environnement **MQSERVER** pour définir le canal entre votre machine IBM MQ MQI client et une machine serveur, il s'agit du seul canal disponible pour votre application et aucune référence n'est faite à la table de définition de canal du client (CCDT).

Si la demande MQCONN ou MQCONNX spécifie un gestionnaire de files d'attente autre que celui auquel le programme d'écoute est connecté, ou si le paramètre **MQSERVER** *TransportType* n'est pas reconnu, la demande MQCONN ou MQCONNX échoue avec le code retour MQRC_Q_MGR_NAME_ERROR.

Linux **AIX** Sous AIX and Linux, vous pouvez définir **MQSERVER** comme dans l'un des exemples suivants:

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56(2002)'  
export MQSERVER=CHANNEL1/LU62/BOX99
```

Toutes les demandes MQCONN ou MQCONNX tentent ensuite d'utiliser le canal que vous avez défini, sauf si une structure MQCD a été référencée à partir de la structure MQCNO fournie à MQCONNX, auquel cas le canal spécifié par la structure MQCD est prioritaire sur tout canal spécifié par la variable d'environnement **MQSERVER**.

La variable d'environnement **MQSERVER** est prioritaire sur toute définition de canal du client pointée par les variables d'environnement **MQCHLLIB** et **MQCHLTAB**.

Procédure

- En fonction de votre plateforme, utilisez l'une des commandes suivantes pour spécifier la définition de canal avec **MQSERVER**.

- Windows** Sous Windows, spécifiez une définition de canal simple comme suit:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName
```

Exemple :

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)'
```

- Linux** **AIX** Sous AIX and Linux, spécifiez une définition de canal simple comme suit:

```
export MQSERVER=ChannelName/TransportType/ConnectionName
```

Exemple :

```
SET MQSERVER=SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)
```

- IBM i** Sous IBM i, spécifiez une définition de canal simple comme suit:

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('ChannelName/TransportType/ConnectionName')
```

Exemple :

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)')
```

Remarques :

- Le *ChannelName* doit avoir le même nom que celui défini sur le serveur. Il ne peut pas contenir de barre oblique (/) car ce caractère est utilisé pour séparer le nom de canal, le type de transport et le nom de connexion. Lorsque la variable d'environnement **MQSERVER** est utilisée pour définir un canal client, une longueur de message maximale (**MAXMSGL**) de 100 Mo est utilisée. Par conséquent, la taille de message maximale en vigueur pour le canal est la valeur spécifiée dans le canal SVRCONN sur le serveur.
- TransportType* peut être l'un des types LU62, TCP, NETBIOS, SPX, selon votre plateforme client IBM MQ.
- Linux** **AIX** Sous AIX and Linux, *TransportType* est sensible à la casse et doit être en majuscules. Un appel MQCONN ou MQCONNX renvoie 2058 si le type de transport n'est pas reconnu

- *ConnectionName* est le nom du serveur tel qu'il est défini pour le protocole de communication (*TransportType*). Il doit s'agir d'un nom de réseau qualifié complet, par exemple AMACHINE.ACOMPANY.COM(1414).
- *ConnectionName* peut être une liste de noms de connexion séparés par des virgules. Les noms de connexion de la liste sont utilisés de la même manière que plusieurs connexions dans une table de connexion client. La liste de noms de connexion peut être utilisée comme alternative aux groupes de gestionnaires de files d'attente pour spécifier plusieurs connexions que le client doit essayer. Si vous configurez un gestionnaire de files d'attente multi-instance, vous pouvez utiliser une liste de noms de connexion pour spécifier des instances de gestionnaire de files d'attente différentes.
- Pour annuler **MQSERVER** et revenir à la table de définition de canal du client indiquée par **MQCHLLIB** et **MQCHLTAB**, entrez la commande suivante:

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

```
unset MQSERVER
```

- **Windows** Sous Windows :

```
SET MQSERVER=
```

Exemple

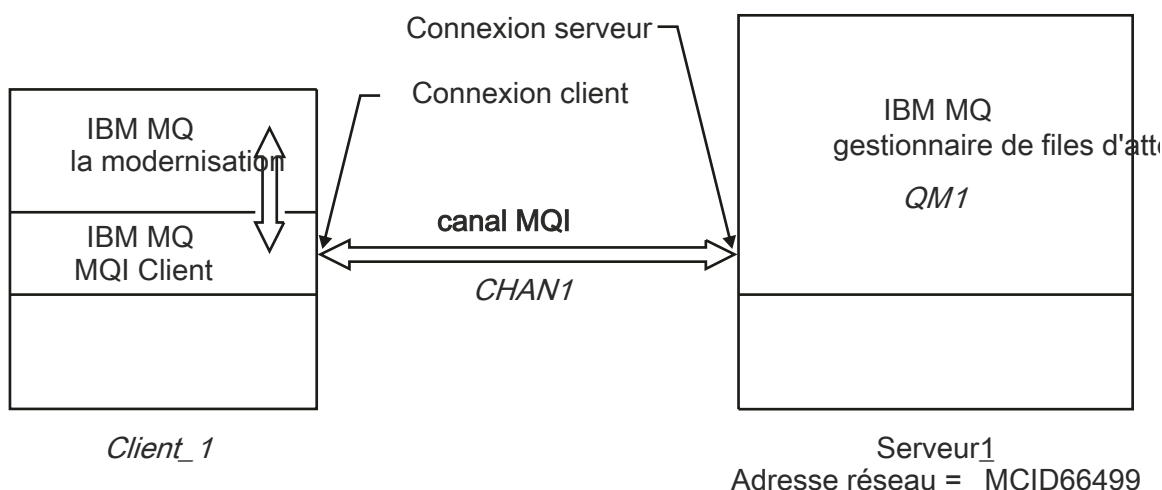


Figure 1. Exemple de définition de canal simple

Pour créer la définition de canal simple illustrée dans la [Figure 1](#), à la page 41, utilisez les commandes suivantes:

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

```
export MQSERVER=CHANNEL1/TCP/'MCID66499'
```

- **Windows** Sous Windows :

```
SET MQSERVER=CHANNEL1/TCP/MCID66499
```

Remarque : Pour plus d'informations sur la modification du numéro de port TCP/IP, voir [«Modification du port TCP/IP par défaut»](#), à la page 42.

Voici d'autres exemples de définitions de canal simples:

- **Windows** Sous Windows :

```
SET MQSERVER=CHANNEL1/TCP/9.20.4.56
SET MQSERVER=CHANNEL1/NETBIOS/BOX643
```

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

```
export MQSERVER=CHANNEL1/TCP/'9.20.4.56'
export MQSERVER=CHANNEL1/LU62/BOX99
```

où BOX99 est la LU 6.2 ConnectionName.

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('CHANNEL1/TCP/9.20.4.56(1416)')
```

Sur le IBM MQ MQI client, toutes les demandes **MQCONN** ou **MQCONNX** tentent ensuite d'utiliser le canal que vous avez défini, sauf si le canal est remplacé dans une structure MQCD référencée à partir de la structure MQCNO fournie à **MQCONNX**.

Tâches associées

«Utilisation des variables d'environnement IBM MQ», à la page 67

Vous pouvez utiliser des commandes pour afficher les paramètres en cours ou pour réinitialiser les valeurs des variables d'environnement IBM MQ .

«Création d'un canal de connexion client sur le IBM MQ MQI client à l'aide de MQCNO», à la page 43

Vous pouvez définir un canal de connexion client sur le poste de travail client à l'aide de la structure MQCNO sur un appel MQCONNX.

Modification du port TCP/IP par défaut

Par défaut, pour TCP/IP, IBM MQ suppose que le canal sera connecté au port 1414. Si nécessaire, vous pouvez modifier la valeur par défaut.

Pourquoi et quand exécuter cette tâche

Vous pouvez modifier le numéro de port à l'aide de l'une des trois options suivantes:

- En utilisant la variable d'environnement **MQSERVER** .
- En modifiant le fichier `mqclient.ini` .
- En ajoutant IBM MQ au fichier de services.

Procédure

- Pour modifier le numéro de port à l'aide de la variable d'environnement **MQSERVER** , ajoutez le numéro de port entre crochets en tant que dernière partie de *ConnectionName*, par exemple:

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

```
export MQSERVER='ChannelName/TransportType/ConnectionName(PortNumber)'
```

- **Windows** Sous Windows :

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(PortNumber)
```

- Pour modifier le numéro de port à l'aide du fichier `mq.ini`, ajoutez le numéro de port au nom de protocole, par exemple:

```
TCP:
port=2001
```

- Pour modifier le numéro de port en ajoutant IBM MQ au fichier de services, suivez les étapes décrites dans «Utilisation du programme d'écoute TCP/IP sous AIX and Linux», à la page 289.

Windows *Modification du socket par défaut SPX*

Par défaut, pour SPX, IBM MQ suppose que le canal sera connecté au socket 5E86. Si nécessaire, vous pouvez modifier la valeur par défaut.

Pourquoi et quand exécuter cette tâche

Vous pouvez modifier le numéro de port à l'aide de l'une des options suivantes:

- En utilisant la variable d'environnement **MQSERVER**.

Pour les connexions SPX, spécifiez `ConnectionName` et le socket sous la forme `network.node(socket)`. Si le client et le serveur IBM MQ se trouvent sur le même réseau, le réseau n'a pas besoin d'être spécifié. Si vous utilisez le socket par défaut, il n'est pas nécessaire de le spécifier.

- En modifiant la strophe SPX du fichier `mqclient.ini` file.Changing du fichier `qm.ini`.

Procédure

- Pour modifier le numéro de port d'une connexion SPX à l'aide de la variable d'environnement **MQSERVER**, spécifiez `ConnectionName` et le socket sous la forme `network.node(socket)`, comme illustré dans l'exemple suivant:

```
SET MQSERVER=ChannelName/TransportType/ConnectionName(SocketNumber)
```

Remarque : Si le client et le serveur IBM MQ se trouvent sur le même réseau, vous n'avez pas besoin de spécifier le réseau. Si vous utilisez le socket par défaut, vous n'avez pas besoin de le spécifier.

- Pour modifier le numéro de port à l'aide du fichier `qm.ini`, ajoutez le numéro de port au nom de protocole, par exemple:

```
SPX:
socket=5E87
```

Création d'un canal de connexion client sur le IBM MQ MQI client à l'aide de MQCNO

Vous pouvez définir un canal de connexion client sur le poste de travail client à l'aide de la structure MQCNO sur un appel MQCONN.

Pourquoi et quand exécuter cette tâche

Une application IBM MQ MQI client peut utiliser la structure d'options de connexion, MQCNO, sur un appel **MQCONN** pour référencer une structure de définition de canal, MQCD, qui contient la définition d'un canal de connexion client.

De cette manière, l'application client peut spécifier les attributs **ChannelName**, **TransportType** et **ConnectionName** d'un canal lors de l'exécution, ce qui permet à l'application client de se connecter simultanément à plusieurs gestionnaires de files d'attente de serveur.

Notez que si vous définissez un canal à l'aide de la variable d'environnement **MQSERVER**, il n'est pas possible de spécifier les attributs **ChannelName**, **TransportType** et **ConnectionName** lors de l'exécution.

Une application client peut également spécifier les attributs d'un canal tel que **MaxMsgLength** et **SecurityExit**. La spécification de ces attributs permet à l'application client de spécifier des valeurs pour les attributs qui ne sont pas les valeurs par défaut et permet aux programmes d'exit de canal d'être appelés à l'extrémité client d'un canal MQI.

Si un canal utilise le protocole TLS (Transport Layer Security), une application client peut également fournir des informations relatives à TLS dans la structure MQCD. Des informations supplémentaires relatives à TLS peuvent être fournies dans la structure des options de configuration TLS, MQSCO, qui est également référencée par la structure MQCNO sur un appel **MQCONN**.

Pour plus d'informations sur les structures MQCNO, MQCD et MQSCO, voir [MQCNO](#), [MQCD](#) et [MQSCO](#).

Remarque : L'exemple de programme pour MQCONN est appelé **amqscnxc**. Un autre exemple de programme appelé **amqsss1c** illustre l'utilisation de la structure MQSCO.

Tâches associées





«Création d'un canal de connexion client sur le IBM MQ MQI client à l'aide de MQSERVER», à la page 39
Vous pouvez définir un canal de connexion client sur un poste de travail client à l'aide de la variable d'environnement **MQSERVER**.

Création de définitions de connexion serveur et de connexion client sur le serveur

Vous pouvez créer les deux définitions sur le serveur, puis mettre la définition de connexion client à la disposition du client.

Pourquoi et quand exécuter cette tâche

Vous devez d'abord définir un canal de connexion serveur, puis définir un canal de connexion client:

- Sur toutes les plateformes, vous pouvez utiliser des commandes IBM MQ Script (MQSC), des commandes PCF (Programmable Command Format) pour définir un canal de connexion serveur sur la machine serveur.
-   Sous Linux et Windows, vous pouvez également utiliser IBM MQ Explorer.
-  Sous z/OS, vous pouvez également utiliser les panneaux Opération et Contrôle.
-  Sous IBM i, vous pouvez également utiliser l'interface du panneau.

Les définitions de canal de connexion client créées sur le serveur sont mises à la disposition des clients à l'aide d'une table de définition de canal du client (CCDT).

Procédure

1. Pour définir un canal de connexion serveur, voir [«Définition du canal de connexion serveur sur le serveur»](#), à la page 58.
2. Pour définir un canal de connexion client, voir [«Définition du canal de connexion client sur le serveur»](#), à la page 59.

Tâches associées

«Configuration d'une table de définition de canal du client au format binaire», à la page 45

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

[«Définition du canal de connexion serveur sur le serveur», à la page 58](#)

Créez une définition de canal de connexion serveur pour le gestionnaire de files d'attente.

[«Définition du canal de connexion client sur le serveur», à la page 59](#)

Après avoir défini le canal de connexion serveur, vous pouvez définir le canal de connexion client correspondant.

[«Accès aux définitions de canal de connexion client», à la page 60](#)

Vous pouvez mettre la table de définition de canal du client (CCDT) à la disposition des applications client en la copiant ou en la partageant, puis en spécifiant son emplacement et son nom sur l'ordinateur client. Vous pouvez également localiser une table de définition de canal du client (CCDT) via une URL.

Configuration des tables de définition de canal du client


Une table de définition de canal du client (CCDT) définit les canaux de connexion du client et leurs attributs. Les clients lisent ce fichier pour déterminer à quels gestionnaires de files d'attente se connecter. Le fichier CCDT peut être au format JSON ou binaire.

Pourquoi et quand exécuter cette tâche

Le gestionnaire de files d'attente ne lit pas le fichier CCDT. Il est utilisé uniquement pour fournir des définitions de canal et des informations d'authentification aux clients.

Une table de définition de canal du client au format binaire est créée automatiquement lors de la création d'un gestionnaire de files d'attente. Vous mettez à jour les définitions de canal du client stockées dans cette table uniquement à l'aide de la commande **runmqsc**.

Une table de définition de canal du client au format JSON est un fichier texte à plat avec une extension .json. Vous créez et mettez à jour cette table manuellement, ce qui est moins restrictif que l'utilisation de la commande **runmqsc**.

 Les clients z/OS JMS exécutés dans un serveur d'applications utilisent une table de définition de canal du client (CCDT) pour référencer les détails de connexion du gestionnaire de files d'attente éloignées. Depuis la IBM MQ for z/OS 9.1, IBM MQ Advanced for z/OS Value Unit Edition permet aux clients JMS de se connecter à distance aux gestionnaires de files d'attente sur d'autres partitions logiques z/OS. Par conséquent, ces clients peuvent également utiliser des tables de définition de canal du client.

Pour vous aider à configurer les tables de définition de canal du client afin qu'elles fonctionnent avec vos clients, effectuez l'une des tâches suivantes:

Procédure

- [«Configuration d'une table de définition de canal du client au format binaire», à la page 45](#)
- [«Configuration d'une table de définition de canal du client au format JSON», à la page 48](#)
- [«Emplacements de la table de définition de canal du client», à la page 55](#)
- [«URL d'accès à la table de définition de canal du client», à la page 56](#)

Concepts associés

[MQI Client : Table de définition de canal du client \(CCDT\)](#)

Tâches associées

[«Configuration d'un cluster uniforme», à la page 445](#)

Les clusters uniformes permettent aux applications d'être conçues pour l'échelle et la disponibilité, et peuvent se connecter à n'importe quel gestionnaire de files d'attente au sein de ce cluster uniforme.

Configuration d'une table de définition de canal du client au format binaire

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut

est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

Avant de commencer

Vous pouvez également créer une table de définition de canal du client au format JavaScript Object Notation (JSON), et l'utilisation de ce format alternatif présente certains avantages par rapport à l'utilisation d'une table de définition de canal du client binaire. Voir [«Configuration d'une table de définition de canal du client au format JSON»](#), à la page 48.

Les clients sur toutes les plateformes peuvent afficher et utiliser des tables de définition de canal du client. Toutefois, la table de définition de canal du client binaire ne peut être créée et modifiée que sous IBM MQ for Multiplatforms.

Pourquoi et quand exécuter cette tâche

Multi

Sous [Multiplateformes](#) :

- Une table de définition de canal du client (CCDT) binaire est créée automatiquement dans le répertoire @ipcc sous le répertoire de données du gestionnaire de files d'attente.
- En plus d'être créée automatiquement, la table de définition de canal du client (CCDT) binaire associée à un gestionnaire de files d'attente est synchronisée avec les définitions d'objet. Lorsque vous définissez, modifiez ou supprimez un objet canal client, la définition d'objet de gestionnaire de files d'attente et l'entrée de la table de définition de canal du client sont mises à jour dans le cadre de la même opération.

Remarques :

- La conception du fichier CCDT IBM MQ est que le fichier CCDT est réduit, uniquement après la définition de tous les canaux de connexion client définis par l'utilisateur. Lorsqu'un canal de connexion client est supprimé, il est simplement marqué comme supprimé dans le fichier CCDT, mais il n'est pas physiquement supprimé.
- Pour forcer la réduction du fichier CCDT, après la suppression d'un ou de plusieurs canaux de connexion client, exécutez la commande suivante:

```
rcrmqobj -m QM80 -t clchltab
```

- La commande **runmqsc** permet de modifier l'emplacement et le contenu de la table de définition de canal du client (CCDT) binaire.

Les clients sur toutes les plateformes peuvent afficher et utiliser une table de définition de canal du client binaire.

Procédure

Multi

- Créez une table de définition de canal du client binaire par défaut.

Sous [Multiplateformes](#), une table de définition de canal du client binaire par défaut appelée AMQCLCHL.TAB est créée lorsque vous créez un gestionnaire de files d'attente.

Par défaut, AMQCLCHL.TAB se trouve dans le répertoire suivant sur un serveur:

- **IBM i** Sous IBM i, dans le système de fichiers intégré:

```
/QIBM/UserData/mqm/qmgrs/QUEUENAME/&ipcc
```

- **Linux** **AIX** Sur les systèmes AIX and Linux :

```
/prefix/qmgrs/QUEUENAME/@ipcc
```

Le nom du répertoire référencé par *QUEUEMANAGERNAME* est sensible à la casse sur les systèmes AIX and Linux . Le nom de répertoire peut être différent du nom du gestionnaire de files d'attente, s'il contient des caractères spéciaux.

– **Windows** Sous Windows :

MQ_INSTALLATION_PATH\data\qmgrs*QUEUEMANAGERNAME*\@ipcc

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

Toutefois, vous avez peut-être choisi d'utiliser un répertoire différent pour les données du gestionnaire de files d'attente. Vous pouvez spécifier le paramètre **-md DataPath** lorsque vous utilisez la commande **crtmqm** . Dans ce cas, AMQCLCHL . TAB se trouve dans le répertoire @ipcc du chemin de données *DataPath* que vous avez spécifié.

- Localisez la table de définition de canal du client:

- Sur l'ordinateur client
- A un emplacement partagé par plusieurs clients
- Sur le serveur en tant que fichier partagé

Voir «Emplacements de la table de définition de canal du client», à la page 55.

a) Créez une table de définition de canal du client binaire directement sur une machine client.

- Utilisez la commande **runmqsc** avec le paramètre **-n** .
- La table de définition de canal du client est créée à l'emplacement indiqué par **MQCHLLIB**, et avec le nom de fichier indiqué par **MQCHLTAB**, qui est AMQCLCHL . TAB par défaut.
- **Important:** Si vous spécifiez le paramètre **-n** , vous ne devez spécifier aucun autre paramètre.

b) Modifiez l'emplacement.

Vous pouvez modifier le chemin d'accès à la table de définition de canal du client en définissant **MQCHLLIB**. N'oubliez pas que si vous avez plusieurs gestionnaires de files d'attente sur le même serveur, ils partagent le même emplacement CCDT.

- Accès à la CCDT

Vous pouvez accéder à la table de définition de canal du client:

- A distance à partir d'un fichier, d'une URL ftp ou d'une URL http, en définissant la variable d'environnement **MQCCDTURL** .
- En local, en définissant les variables d'environnement **MQCHLLIB** et **MQCHLTAB** .
- En local, en définissant les attributs **ChannelDefinitionDirectory** et **ChannelDefinitionFile** de la strophe CHANNELS dans le fichier de configuration du client.

Pour obtenir divers exemples, voir «Emplacements de la table de définition de canal du client», à la page 55 .

- Affichez ou éditez le contenu de la table de définition de canal du client.

Vous pouvez afficher le contenu de la table de définition de canal du client à l'aide de la commande **runmqsc** :

1. Définissez les variables d'environnement sur [Accès à la table de définition de canal du client](#)
2. Exécutez la commande **runmqsc -n**
3. Exécutez la commande **DISPLAY CHANNEL (*)**, par exemple

– **Multi** Sous Multiplateformes, vous pouvez également éditer le contenu de la table de définition de canal du client (CCDT) binaire à l'aide de la commande **runmqsc** . Chaque entrée d'une table de définition de canal du client représente une connexion client à un gestionnaire de files d'attente spécifique. Une nouvelle entrée est ajoutée lorsque vous définissez un canal de connexion client à l'aide de la commande **DEFINE CHANNEL** , et l'entrée est mise à jour lorsque vous modifiez les canaux

de connexion client à l'aide de la commande **ALTER CHANNEL** . Pour plus d'exemples d'utilisation de la commande, voir [runmqsc](#) .

- Fournissez aux clients les informations d'authentification pour vérifier la révocation de certificat TLS.
 - a) Définissez une liste de noms contenant des objets d'informations d'authentification.
 - b) Définissez l'attribut de gestionnaire de files d'attente **SSLCRLNL** sur le nom de la liste de noms.

Concepts associés

[Utilisation des certificats révoqués](#)

Tâches associées


«Configuration d'une table de définition de canal du client au format JSON», à la page 48

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Vous utilisez un éditeur de texte pour créer et mettre à jour un objet JSON (JavaScript Object Notation) CCDT.

Configuration d'une table de définition de canal du client au format JSON

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Vous utilisez un éditeur de texte pour créer et mettre à jour un objet JSON (JavaScript Object Notation) CCDT.

Avant de commencer

 Si vous utilisez IBM MQ for Multiplatforms, vous pouvez utiliser à la place la table de définition de canal du client binaire qui est créée automatiquement lorsque vous créez un gestionnaire de files d'attente. Voir «[Configuration d'une table de définition de canal du client au format binaire](#)», à la page 45.

Pourquoi et quand exécuter cette tâche

Le nom de fichier du schéma CCDT pour le format JSON est:

Linux

```
/opt/mqm/lib/ccdt_schema.json
```

Windows




```
C:\Program Files\IBM\MQ\bin\ccdt_schema.json
```

Il n'existe pas de table de définition de canal du client JSON par défaut et IBM MQ ne fournit aucun outil permettant de créer ou d'éditer des tables de définition de canal du client au format JSON. Néanmoins, vous disposez de plus d'options de configuration lorsque vous développez manuellement une table de définition de canal du client JSON que lorsque vous utilisez la commande **runmqsc** pour utiliser une table de définition de canal du client binaire:

- Il n'est pas nécessaire d'utiliser IBM MQ for Multiplatforms pour créer et éditer un fichier CCDT JSON.
- A l'aide du format JSON, vous pouvez définir des définitions de canal en double du même nom. Lorsque vous déployez IBM MQ sur le cloud, vous pouvez l'utiliser pour rendre votre déploiement évolutif et hautement disponible.
- Le fichier JSON est lisible par l'utilisateur, ce qui peut simplifier la configuration du gestionnaire de files d'attente.
- Un format de fichier à plat peut être intégré à:
 - Outils de contrôle des versions pour le suivi de l'historique de la table de définition de canal du client
 - Outils d'automatisation en distribution continue
- Vous n'avez pas besoin d'outils spécialisés pour gérer le fichier CCDT.
- Le fichier est plus petit.
- Ce format offre une compatibilité avec les versions antérieures et antérieures.

Remarques :

1. La norme JSON considère les clés en double comme étant valides. Toutefois, l'analyseur syntaxique JSON utilise uniquement la dernière valeur de lecture des clés en double lors de l'affectation d'attributs. Par conséquent, lors de la définition de canaux en double, chaque canal doit être un élément d'une valeur de tableau affectée à la clé 'channel1'.
2. Les tables de définition de canal du client JSON ne prennent pas en charge le stockage des emplacements de serveur LDAP (Lightweight Directory Access Protocol) pour les listes de révocation de certificat (CRL) et les informations d'emplacement du répondeur OCSP (Online Certificate Status Protocol).

Plateforme	Codage du client JMS	Codage du client C
 IBM i	ASCII	EBCDIC
 AIX, Linux, and Windows	ASCII	ASCII
 z/OS	ASCII ou EBCDIC	Non applicable



Avertissement : Lorsque vous fournissez une définition pour un canal via une table de définition de canal du client JSON (y compris une table de définition de canal *sparse* qui n'inclut pas tous les attributs), une définition de canal complète est construite avec tous les attributs définis, en utilisant les valeurs par défaut pour tout élément non spécifié dans le JSON.

Par conséquent, vous devez fournir des valeurs spécifiques pour chaque attribut pour lequel vous ne souhaitez pas la valeur par défaut.




Procédure

- Créer une table de définition de canal du client JSON
 - a) Créez un fichier à plat avec une extension `.json` avec un éditeur de texte générique.
 - b) Définissez une table de définition de canal du client.

Voir «Exemples CCDT JSON», à la page 52 et «Attributs de canal pris en charge par la table de définition de canal du client JSON», à la page 50.
- Localisez la table de définition de canal du client:
 - Sur l'ordinateur client
 - A un emplacement partagé par plusieurs clients
 - Sur le serveur en tant que fichier partagéVoir «Emplacements de la table de définition de canal du client», à la page 55.
- Validation d'une table de définition de canal du client JSON

Validez la table de définition de canal du client par rapport au schéma avec un éditeur de liens JSON. Voir [Comment valider un fichier JSON CCDT IBM MQ par rapport au schéma](#) pour plus d'informations sur la création d'un fichier CCDT avec deux canaux et la validation de son fonctionnement.

Le schéma CCDT est inclus avec les packages produit et client:

 -   Sur les systèmes AIX and Linux :
`$MQ_INSTALLATION_PATH/lib` et `/lib` dans les packages produit et client respectivement.
 -  Sous Windows :
`%MQ_INSTALLATION_PATH%\bin` et `\bin` dans les packages produit et client respectivement.

Remarques :

- Les pointeurs JSON sont disponibles en ligne.
- Le schéma définit les attributs obligatoires avec la clé 'required'.
- Le schéma définit les types de données d'attribut avec la clé 'type'.
- Accès à la CCDT

Vous pouvez accéder à la table de définition de canal du client:

- A distance à partir d'un fichier, ftp ou d'une URL http, en définissant la variable d'environnement **MQCCDTURL**.
- En local, en définissant les variables d'environnement **MQCHLLIB** et **MQCHLTAB**.
- En local, en définissant les attributs **ChannelDefinitionDirectory** et **ChannelDefinitionFile** de la strophe CHANNELS dans le fichier de configuration du client.

Pour obtenir divers exemples, voir «[Emplacements de la table de définition de canal du client](#)», à la page 55.

- Afficher ou éditer le contenu de la table de définition de canal du client

Chaque entrée d'une table de définition de canal du client représente une connexion client à un gestionnaire de files d'attente spécifique. Vous pouvez afficher ou éditer le contenu de la table de définition de canal du client à l'aide d'un éditeur de texte.

Si vous souhaitez uniquement afficher la table de définition de canal du client, vous pouvez également utiliser la commande **runmqsc** comme suit:

1. Définissez des variables d'environnement pour vous permettre d'accéder à la table de définition de canal du client, comme décrit à l'étape précédente.
2. Exécutez la commande `runmqsc -n`. Pour plus d'informations, voir [runmqsc](#).
3. Exécutez la commande **DISPLAY CHANNEL**. Par exemple, exécutez `DISPLAY CHANNEL(*)`.

Concepts associés

[Utilisation des certificats révoqués](#)

Tâches associées

«[Configuration d'une table de définition de canal du client au format binaire](#)», à la page 45

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

«[Configuration d'un cluster uniforme](#)», à la page 445

Les clusters uniformes permettent aux applications d'être conçues pour l'échelle et la disponibilité, et peuvent se connecter à n'importe quel gestionnaire de files d'attente au sein de ce cluster uniforme.

Attributs de canal pris en charge par la table de définition de canal du client JSON

Liste des attributs de canal de connexion client pris en charge par la table de définition de canal du client JSON. Cette liste est un sous-ensemble des attributs pris en charge par la table de définition de canal du client binaire.

Mappage d'attributs

Ces attributs sont insérés dans l'objet canal suivant:

```
{ "channel": [ { $CHANNEL_1_KEY_VALUE_LIST }, ..., { $CHANNEL_N_KEY_VALUE_LIST } ] }
```

où `$CHANNEL_X_KEY_VALUE_LIST` est une liste séparée par des virgules des attributs répertoriés dans le tableau suivant.

Voir «[Exemples CCDT JSON](#)», à la page 52 pour les cas d'utilisation de base.

Le schéma JSON est fourni dans /opt/mqm/lib/ccdt_schema.json. Pour savoir quelles valeurs sont valides pour chacun des attributs, consultez le schéma JSON.

Le tableau suivant répertorie l'objet JSON, la clé et le type de données, ainsi que la définition d'attribut de canal binaire correspondante.



Avertissement : Les attributs requis sont le canal **name** et le canal **type**. Si vous définissez également **portRange**, les attributs *low* et *high* sont également requis.

Objet JSON	Clé JSON	Type de données JSON	Définition d'attribut binaire
channel (tableau)	Nom	STRING	Canal
channel (tableau)	type	STRING	CHLTYPE
channel.clientConnection	queueManager	STRING	QMNAME
channel.clientConnection.connection (tableau)	host	STRING	CONNNAME
channel.clientConnection.connection	port	INT	CONNNAME
channel.compression.header (tableau)	en-tête	STRING	COMPHDR
channel.compression.message (tableau)	Message	STRING	COMPMSG
channel.connectionManagement	affinity	STRING	AFFINITY
channel.connectionManagement	clientWeight	INT	CLNTWGHT
channel.connectionManagement	defaultReconnect	STRING	DEFRECON
channel.connectionManagement	disconnectInterval	INT	DISCINT
channel.connectionManagement	heartInterval	INT	HBINT
channel.connectionManagement	KeepAliveInterval	INT	KAINT
channel.connectionManagement	sharingConversations	INT	SHARECNV
channel.connectionManagement.localAddress (tableau)	host	STRING	LOCLADDR
channel.connectionManagement.localAddress (tableau)	port	INT	LOCLADDR
channel.connectionManagement.localAddress.portRange	haut	INT	LOCLADDR
channel.connectionManagement.localAddress.portRange	bas	INT	LOCLADDR
channel.exits.receive (tableau)	Nom	STRING	RCVEXIT
channel.exits.receive (tableau)	userData	STRING	RCVDATA
channel.exits.security	Nom	STRING	SCYEXIT
channel.exits.security	userData	STRING	SCYDATA
channel.exits.send (tableau)	Nom	STRING	SENDEXIT
channel.exits.send (tableau)	userData	STRING	SENDATA
channel.general	description	STRING	DESCR
channel.general	maximumMessageLen gth	INT	MAXMSGL

Objet JSON	Clé JSON	Type de données JSON	Définition d'attribut binaire
channel.timestamps	Modifié	STRING	ALTDATA et ALTTIME
channel.transmissionSecurity	certificateLabel	STRING	CERTLABL
channel.transmissionSecurity	Nom certificatePeer	STRING	SSLPEER
channel.transmissionSecurity	cipherSpecification	STRING	SSLCIPH

Remarques :

- `channel.connectionManagement.localAddress` peut être défini comme l'une des combinaisons de touches suivantes:
 - Hôte et port
 - host et portRange
 - port
 - portRange
- La clé JSON `channel.timestamps.altered` est facultative et, si elle n'est pas définie, la valeur par défaut est l'heure de la dernière modification du fichier CCDT JSON. Toutefois, si l'environnement est configuré pour extraire la table de définition de canal du client à partir d'une URL, la valeur par défaut correspond à l'heure à laquelle le fichier a été téléchargé pour la dernière fois.
- `channel.clientConnection.connection` doit inclure les clés d'hôte et de port.
- La clé modifiée est une chaîne unique qui encapsule les attributs ALTDATA et ALTTIME.
- Le type de transport ne peut être que TCP. Par conséquent, les attributs suivants ne sont pas définis dans le schéma:
 - **TRPTYPE**
 - **USERID**
 - **PASSWORD**
 - **MODENAME**
 - **TPNAME**

Référence associée

[Attributs de canal pour les types de canal](#)

Exemples CCDT JSON

Utilisez les exemples répertoriés dans cette rubrique comme base pour vos besoins.

Ouvrez un éditeur de texte générique et copiez l'un des exemples suivants:

- [«Définition d'une connexion client simple»](#), à la page 52
- [«Définition d'un canal et d'un gestionnaire de files d'attente à l'aide de TLS»](#), à la page 53
- [«Définir un canal et un gestionnaire de files d'attente n'utilisant pas TLS»](#), à la page 53
- [«Définir deux canaux portant le même nom»](#), à la page 53
- [«Liste complète des définitions d'attribut de canal CCDT pour un canal de connexion client»](#), à la page 54

Définition d'une connexion client simple

```
{
  "channel":
```

```
[
  {
    "general":
    {
      "description": "a channel"
    },
    "name": "channel",
    "clientConnection":
    {
      "connection":
      [
        {
          "host": "localhost",
          "port": 1414
        }
      ],
      "queueManager": "QM1"
    },
    "type": "clientConnection"
  }
]
```

Définition d'un canal et d'un gestionnaire de files d'attente à l'aide de TLS

```
{
  "channel": [
    {
      "name": "SSL.SVRCONN",
      "clientConnection": {
        "connection": [
          {
            "host": "aztlan1.fyre.ibm.com",
            "port": 1419
          }
        ],
        "queueManager": "QM92TLS"
      },
      "transmissionSecurity":
      {
        "cipherSpecification": "TLS_AES_128_GCM_SHA256",
        "certificateLabel": "ibmwebspheremqadministrator",
      },
      "type": "clientConnection"
    }
  ]
}
```

Définir un canal et un gestionnaire de files d'attente n'utilisant pas TLS

```
{
  "channel": [
    {
      "name": "SYSTEM.DEF.SVRCONN",
      "clientConnection": {
        "connection": [
          {
            "host": "aztlan1.fyre.ibm.com",
            "port": 1414
          }
        ],
        "queueManager": "QM92"
      },
      "type": "clientConnection"
    }
  ]
}
```

Définir deux canaux portant le même nom

Chaque canal se connecte à deux gestionnaires de files d'attente distincts:

```
{
```

```

"channel":
[
  {
    "general":
    {
      "description": "First channel"
    },
    "name": "channel",
    "clientConnection":
    {
      "connection":
      [
        {
          "host": "localhost",
          "port": 1414
        }
      ],
      "queueManager": "QM1"
    },
    "type": "clientConnection"
  },
  {
    "general":
    {
      "description": "Second channel"
    },
    "name": "channel",
    "clientConnection":
    {
      "connection":
      [
        {
          "host": "localhost",
          "port": 1415
        }
      ],
      "queueManager": "QM2"
    },
    "type": "clientConnection"
  }
]
}

```

Liste complète des définitions d'attribut de canal CCDT pour un canal de connexion client

```

{
  "channel":
  [
    {
      "compression":
      {
        "header": [ "system" ],
        "message": [ "zlibfast" ]
      },
      "connectionManagement":
      {
        "sharingConversations": 10,
        "clientWeight": 1,
        "affinity": "none",
        "defaultReconnect": "yes",
        "heartbeatInterval": 600,
        "keepAliveInterval": -1,
        "localAddress":
        [
          {
            "portRange":
            {
              "low": 2020,
              "high": 3030
            }
          }
        ]
      }
    },
    "exits":
    {
      "receive":
      [

```

```

    {
      "name": "",
      "userData": ""
    }
  ],
  "security":
  {
    "name": "",
    "userData": ""
  },
  "send":
  [
    {
      "name": "",
      "userData": ""
    }
  ]
},
"general":
{
  "description": "First channel",
  "maximumMessageLength": 4194304
},
"name": "the_channel",
"clientConnection":
{
  "connection":
  [
    {
      "host": "localhost",
      "port": 1414
    }
  ],
  "queueManager": "QM1"
},
"timestamps":
{
  "altered": "2018-12-04T15:37:22.000Z"
},
"transmissionSecurity":
{
  "cipherSpecification": "",
  "certificateLabel": "",
  "certificatePeerName": ""
},
"type": "clientConnection"
}
]
}

```

Référence associée

[Attributs de canal pour les types de canal](#)

[Attributs de canal par ordre alphabétique](#)

Emplacements de la table de définition de canal du client

IBM MQ prend en charge l'extraction d'une table de définition de canal du client à partir d'un fichier, d'un FTP ou d'une URL HTTP. Vous pouvez rendre la table de définition de canal du client accessible au client en tant que fichier partagé, alors qu'elle reste sur le serveur. Vous pouvez également distribuer la table de définition de canal du client, soit en copiant la table de définition de canal du client sur des ordinateurs client individuels, soit en copiant la table de définition de canal du client dans un emplacement partagé par plusieurs clients.

Si vous utilisez FTP pour copier le fichier, utilisez l'option `bin` pour définir le mode binaire ; n'utilisez pas le mode ASCII par défaut. Quelle que soit la méthode choisie pour rendre la table de définition de canal du client disponible, l'emplacement doit être sécurisé pour empêcher les modifications non autorisées apportées aux canaux.

Comment héberger le fichier CCDT sur un serveur

La table de définition de canal du client peut être hébergée dans un emplacement central accessible via une URL, ce qui élimine la nécessité de mettre à jour individuellement la table de définition de canal du client pour chaque client déployé. Les applications Native (C/C + +, COBOL et RPG) et .NET non gérées

peuvent extraire la table de définition de canal du client à partir d'une URL, qu'il s'agisse d'un fichier local, d'une ressource FTP ou HTTP.

Le comportement de mise en cache par défaut des clients IBM MQ est qu'un fichier CCDT n'est extrait que si l'heure de modification du fichier est différente de la dernière fois où il a été extrait. Comme pour la plupart des options de configuration client, l'emplacement de l'URL peut être fourni de différentes manières:

- **CCDTUr1Ptr** et **CCDTUr1offset** via la structure MQCNO transmise à l'appel MQCONNX MQI
- Variable d'environnement **MQCCDTURL**
- Attribut **ChannelDefinitionDirectory** dans la strophe Channels de `mqclient.ini`

Les URL authentifiées et non authentifiées sont prises en charge. Voici quelques exemples :

```
export MQCCDTURL=ftp://myuser:password@myhost.sample.com//var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
```

```
export MQCCDTURL=http://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc/AMQCLCHL.TAB
```

Si vous souhaitez utiliser cette prise en charge avec FTP ou HTTP, vous devez toujours héberger le fichier CCDT sur un serveur, mais toutes vos applications client peuvent sélectionner automatiquement les modifications apportées aux définitions de canal sans avoir à envoyer manuellement les mises à jour ou à monter un système de fichiers en réseau sur chaque client. Pour plus d'informations, voir [«URL d'accès à la table de définition de canal du client»](#), à la page 56.

Comment spécifier l'emplacement de la table de définition de canal du client sur le client

Sur un système client, vous pouvez spécifier l'emplacement de la table de définition de canal du client de l'une des manières suivantes:

- Utilisation des variables d'environnement **MQCHLLIB** pour spécifier le répertoire dans lequel se trouve la table et **MQCHLTAB** pour spécifier le nom de fichier de la table.
- Utilisation du fichier de configuration du client. Dans la section CHANNELS, utilisez l'attribut **ChannelDefinitionDirectory** pour indiquer le répertoire dans lequel se trouve la table et l'attribut **ChannelDefinitionFile** pour indiquer le nom de fichier.
- En fournissant une URL (fichier, FTP ou HTTP) pour une table de définition de canal du client qui est hébergée dans un emplacement central, comme décrit précédemment.

Si l'emplacement est spécifié à la fois dans le fichier de configuration du client et à l'aide de variables d'environnement, les variables d'environnement sont prioritaires. Vous pouvez utiliser cette fonction pour spécifier un emplacement standard dans le fichier de configuration du client et le remplacer à l'aide de variables d'environnement si nécessaire.

Si vous utilisez une URL pour indiquer l'emplacement de la table de définition de canal du client, l'ordre de priorité d'une application client native pour trouver la définition de canal du client est celui décrit dans [«URL d'accès à la table de définition de canal du client»](#), à la page 56.

URL d'accès à la table de définition de canal du client

Vous pouvez héberger une table de définition de canal du client (CCDT) dans un emplacement central accessible via une URL, ce qui supprime la nécessité de mettre à jour individuellement la CCDT pour chaque client déployé.

Une table de définition de canal du client peut être référencée par une adresse URL de l'une des façons suivantes :

- Par programmation avec MQCNO
- A l'aide de variables d'environnement



Avertissement : Vous pouvez utiliser l'option de variable d'environnement pour fournir l'URL uniquement pour les programmes natifs qui se connectent en tant que clients, c'est-à-dire les

applications C, COBOL ou C + +. Les variables d'environnement n'ont pas d'incidence pour Java, JMS ou les applications .NET gérées.

IBM MQ prend en charge l'extraction d'une table de définition de canal du client à partir d'un fichier, d'un FTP ou d'une URL HTTP.

- A l'aide de la section Canaux du fichier `mqcClient.ini`.

La variable d'environnement **MQCCDTURL** vous permet de fournir une URL de fichier, ftp ou http sous la forme d'une valeur unique à partir de laquelle une table de définition de canal du client peut être obtenue.

Vous pouvez également utiliser le chemin de répertoire spécifié par la variable d'environnement **MQCHLLIB** (ou le chemin spécifié par l'attribut **ChannelDefinitionDirectory** dans «Strophe CHANNELS du fichier de configuration client», à la page 193) pour localiser un fichier CCDT, via un fichier, ftp ou une URL http, en plus du répertoire du système de fichiers local existant, à savoir `/var/mqm`. Notez qu'une valeur **MQCHLLIB** est un radical de répertoire et qu'elle fonctionne en combinaison avec **MQCHLTAB** pour dériver l'URL qualifiée complète.

L'authentification de base sur les connexions est prise en charge via les données d'identification codées dans l'URL:

Connexions authentifiées

```
export MQCHLLIB=ftp://myuser:password@myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLLIB=http://myuser:password@myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
```

Connexions non authentifiées

```
export MQCHLLIB=ftp://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLLIB=http://myhost.sample.com/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLLIB=file:///var/mqm/qmgrs/QMGR/@ipcc
```

Remarque : Si vous souhaitez utiliser des connexions authentifiées, vous devez, comme avec JMS, fournir le nom d'utilisateur et le mot de passe codés dans l'URL.

L'ordre de priorité, pour une application client native, de la recherche d'une définition de canal client est désormais:

1. MQCD fourni par **ClientConnOffset** et **ClientConnPtr** dans MQCNO.
2. URL fournie par **CCDTUrlOffset** et **CCDTUrlPtr** dans MQCNO.
3. Variable d'environnement **MQSERVER**.
4. Si un fichier `mqcClient.ini` est défini et que la strophe Channels contient un attribut **ServerConnectionParms**, le canal qu'il définit est utilisé. Pour plus d'informations, voir «Fichier de configuration IBM MQ MQI client, `mqcClient.ini`», à la page 174 et «Strophe CHANNELS du fichier de configuration client», à la page 193.
5. Variable d'environnement **MQCCDTURL**.
6. Variable d'environnement **MQCHLLIB** et **MQCHLTAB**.
7. **ChannelDefinitionDirectory** et **ChannelDefinitionFile** dans «Strophe CHANNELS du fichier de configuration client», à la page 193.

Important : L'accès à un fichier CCDT à l'aide d'une URL ouvre toujours une copie en lecture seule du fichier, même si vous utilisez le protocole `file://`.

La tentative d'ouverture d'un fichier CCDT pour l'accès en écriture, par exemple lors de l'utilisation de la commande **DEFINE CHANNEL MQSC** à partir d'un client, renvoie un message d'erreur indiquant que le fichier n'a pas pu être ouvert pour l'accès en écriture.

Il est toutefois possible de lire les définitions d'informations de canal et d'authentification à l'aide de **runmqsc**.

Tâches associées

«Accès aux définitions de canal de connexion client», à la page 60

Vous pouvez mettre la table de définition de canal du client (CCDT) à la disposition des applications client en la copiant ou en la partageant, puis en spécifiant son emplacement et son nom sur l'ordinateur client. Vous pouvez également localiser une table de définition de canal du client (CCDT) via une URL.

«[Configuration d'une table de définition de canal du client au format binaire](#)», à la page 45

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

[Utilisation d'une table de définition de canal du client avec IBM MQ classes for JMS](#)

Référence associée

[CCDTURL](#)

[MQCNO-Options de connexion](#)

[XMSC_WMQ_CCDTURL](#)

Windows

Canaux de connexion client dans Active Directory

Sur les systèmes Windows qui prennent en charge Active Directory, IBM MQ publie les canaux de connexion client dans Active Directory pour fournir une liaison client-serveur dynamique.

Lorsque des objets de canal de connexion client sont définis, ils sont écrits dans un fichier de définition de canal client, appelé AMQCLCHL .TAB par défaut. Si les canaux de connexion client utilisent le protocole TCP/IP, le serveur IBM MQ les publie également dans Active Directory. Lorsque le client IBM MQ détermine comment se connecter au serveur, il recherche une définition d'objet de canal de connexion client appropriée à l'aide de l'ordre de recherche suivant:

1. [MQCONN](#) Structure de données MQCD
2. Variable d'environnement [MQSERVER](#)
3. Fichier de définition de canal du client
4. Active Directory

Cet ordre signifie que les applications en cours ne sont affectées par aucune modification. Vous pouvez considérer ces entrées dans Active Directory comme des enregistrements dans le fichier de définition de canal du client et le client IBM MQ les traite de la même manière. Pour configurer et administrer la prise en charge de la publication des définitions de canal de connexion client dans Active Directory, utilisez la commande `setmqscp`, comme décrit dans [setmqscp](#).

Définition du canal de connexion serveur sur le serveur

Créez une définition de canal de connexion serveur pour le gestionnaire de files d'attente.

Procédure

1. Sur la machine serveur, définissez un canal avec le nom de votre choix et le type de canal *server-connection*.

Exemple :

```
DEFINE CHANNEL(CHAN2) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
DESCR('Server-connection to Client_2')
```

2. Utilisez la commande suivante pour autoriser l'accès de connexion entrante à votre gestionnaire de files d'attente:

```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP address') MCAUSER('userid')
```

- Où **SET CHLAUTH** utilise le nom du canal défini à l'étape précédente.

- Où '*Adresse IP*' est l'adresse IP du client.
- Où '*userid*' est l'ID que vous souhaitez fournir au canal pour le contrôle d'accès aux files d'attente cible. Cette zone est sensible à la casse.

Vous pouvez choisir d'identifier votre connexion entrante à l'aide d'un certain nombre d'attributs différents. L'exemple utilise l'adresse IP. Les autres attributs incluent l'ID utilisateur du client et le nom distinctif du sujet TLS. Pour plus d'informations, voir [Enregistrements d'authentification de canal](#)

Cette définition de canal est associée au gestionnaire de files d'attente en cours d'exécution sur le serveur.

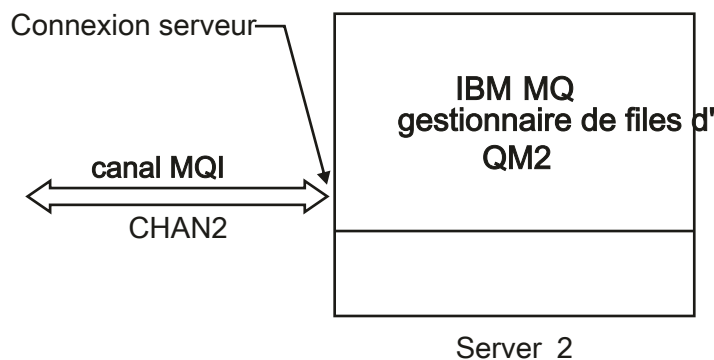


Figure 2. Définition du canal de connexion serveur

Tâches associées

«Définition du canal de connexion client sur le serveur», à la page 59

Après avoir défini le canal de connexion serveur, vous pouvez définir le canal de connexion client correspondant.

Définition du canal de connexion client sur le serveur

Après avoir défini le canal de connexion serveur, vous pouvez définir le canal de connexion client correspondant.

Avant de commencer

Définissez le canal de connexion serveur. Pour plus d'informations, voir [«Définition du canal de connexion serveur sur le serveur»](#), à la page 58.

Procédure

1. Définissez un canal portant le même nom que le canal de connexion serveur, mais dont le type de canal est *client-connection*. Vous devez indiquer le nom de la connexion (CONNNAME). Pour TCP/IP, le nom de connexion est l'adresse réseau ou le nom d'hôte de la machine serveur. Il est également conseillé de spécifier le nom du gestionnaire de files d'attente (QMNAME) auquel vous souhaitez que votre application IBM MQ, exécutée dans l'environnement client, se connecte. En faisant varier le nom du gestionnaire de files d'attente, vous pouvez définir un ensemble de canaux pour vous connecter à différents gestionnaires de files d'attente.

```
DEFINE CHANNEL(CHAN2) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
  CONNNAME(9.20.4.26) QMNAME(QM2) DESCR('Client-connection to Server_2')
```

2. Utilisez la commande suivante pour autoriser l'accès de connexion entrante à votre gestionnaire de files d'attente:

```
SET CHLAUTH(CHAN2) TYPE(ADDRESSMAP) ADDRESS('IP-address') MCAUSER('userid')
```

- où la commande **SET CHLAUTH** utilise le nom du canal défini à l'étape précédente.

- Où 'Adresse IP' est l'adresse IP du client.
- Où 'userid' est l'ID que vous souhaitez fournir au canal pour le contrôle d'accès aux files d'attente cible. Cette zone est sensible à la casse.

Vous pouvez choisir d'identifier votre connexion entrante à l'aide d'un certain nombre d'attributs différents. L'exemple utilise l'adresse IP. Les autres attributs incluent l'ID utilisateur du client et le nom distinctif du sujet TLS. Pour plus d'informations, voir [Enregistrements d'authentification de canal](#)

Résultats

Multi Sous [Multiplateformes](#), cette définition de canal est stockée dans un fichier appelé table de définition de canal du client (CCDT), qui est associé au gestionnaire de files d'attente. La table de définition de canal du client peut contenir plusieurs définitions de canal de connexion client.

Pour plus d'informations sur la table de définition de canal du client et sur la manière dont les définitions de canal de connexion client sont stockées dans z/OS, voir [«Configuration d'une table de définition de canal du client au format binaire»](#), à la page 45.

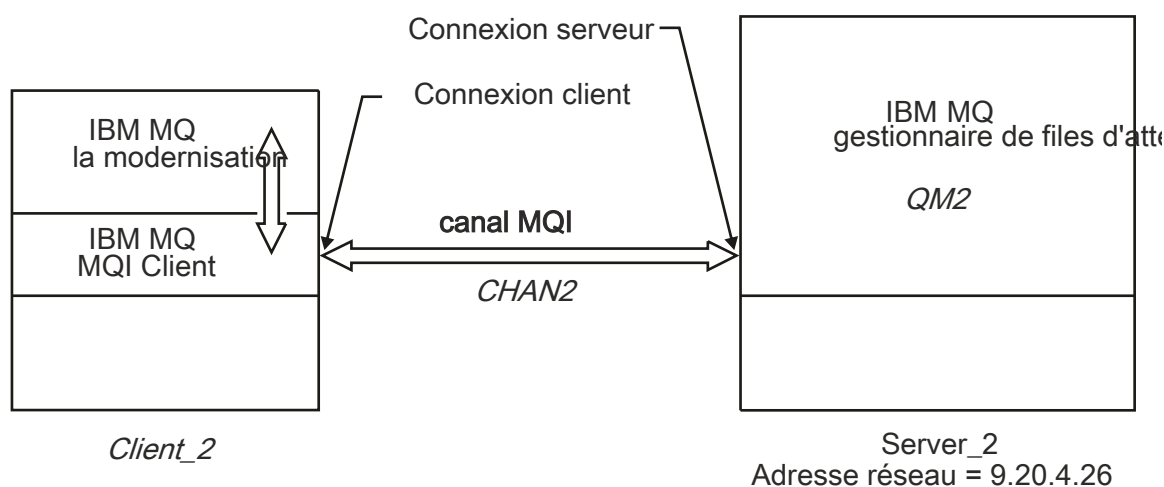


Figure 3. Définition du canal de connexion client

Référence associée

[DEFINE CHANNEL](#) (définition d'un nouveau canal)

[SET CHLAUTH](#) (création ou modification d'un enregistrement d'authentification de canal)

Accès aux définitions de canal de connexion client

Vous pouvez mettre la table de définition de canal du client (CCDT) à la disposition des applications client en la copiant ou en la partageant, puis en spécifiant son emplacement et son nom sur l'ordinateur client. Vous pouvez également localiser une table de définition de canal du client (CCDT) via une URL.

Avant de commencer

Cette tâche suppose que vous avez défini, dans une table de définition de canal du client, les canaux de connexion client dont vous avez besoin. Voir [«Configuration des tables de définition de canal du client»](#), à la page 45.

Pourquoi et quand exécuter cette tâche

Pour qu'une application client utilise la table de définition de canal du client (CCDT), vous devez mettre cette dernière à sa disposition et spécifier son emplacement et son nom. Il existe plusieurs façons de procéder:

- Vous pouvez copier la table de définition de canal du client sur l'ordinateur client.
- Vous pouvez copier la table de définition de canal du client dans un emplacement partagé par plusieurs clients.
- Vous pouvez rendre la table de définition de canal du client accessible au client en tant que fichier partagé, alors qu'elle reste sur le serveur.

Les applications IBM MQ natives (C/C + +, COBOL et RPG) et .NET non gérées peuvent extraire la table de définition de canal du client (CCDT) hébergée dans un emplacement central à partir d'une URL, qu'il s'agisse d'un fichier local, d'une ressource FTP ou d'une ressource http.

Procédure

1. Mettez la table de définition de canal du client à la disposition des applications client de l'une des manières suivantes:
 - a) Facultatif : Copiez la table de définition de canal du client sur l'ordinateur client.
 - b) Facultatif : Copiez la table de définition de canal du client dans un emplacement partagé par plusieurs clients.
 - c) Facultatif : Laissez la table de définition de canal du client sur le serveur, mais rendez cette dernière partageable par le client.
 - d) Facultatif : Définissez un fichier local, une URL ftp ou http pour une table de définition de canal du client hébergée dans un emplacement central de sorte que les applications .NET natives (C/C + +, COBOL et RPG) et non gérées puissent extraire la table de définition de canal du client de cette URL.


Quel que soit l'emplacement choisi pour la table de définition de canal du client, l'emplacement doit être sécurisé pour empêcher les modifications non autorisées apportées aux canaux.

2. Sur le client, spécifiez l'emplacement et le nom du fichier contenant la table de définition de canal du client de l'une des trois manières suivantes:
 - a) Facultatif : Utilisez la strophe CHANNELS du fichier de configuration du client. Pour plus d'informations, voir «Strophe CHANNELS du fichier de configuration client», à la page 193.
 - b) Facultatif : Utilisez les variables d'environnement **MQCHLLIB** et **MQCHLTAB**.

Par exemple, vous pouvez définir les variables d'environnement en entrant:

-   Sur les systèmes AIX and Linux :

```
export MQCHLLIB= MQ_INSTALLATION_PATH/qmgrs/ QUEUEMANAGERNAME /@ipcc
export MQCHLTAB=AMQCLCHL.TAB
```

-  Sur les systèmes IBM i :

```
ADDENVVAR ENVVAR(MQCHLLIB) VALUE('/QIBM/UserData/mqm/qmgrs/QUEUEMANAGERNAME/@ipcc')
ADDENVVAR ENVVAR(MQCHLTAB) VALUE(AMQCLCHL.TAB)
```

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

- c) Facultatif : Sous Windows uniquement, utilisez la commande de contrôle **setmqscp** pour publier les définitions de canal de connexion client dans Active Directory.
- d) Indiquez l'emplacement d'une table de définition de canal du client (CCDT) hébergée de manière centrale via une URL, soit par programmation à l'aide de MQCNO, à l'aide de variables d'environnement, soit à l'aide de sections de fichier *mqclient.ini*. Pour plus d'informations, reportez-vous aux sections «Emplacements de la table de définition de canal du client», à la page 55 et «URL d'accès à la table de définition de canal du client», à la page 56.

Si la variable d'environnement **MQSERVER** est définie, un client IBM MQ utilise la définition de canal de connexion client spécifiée par **MQSERVER** de préférence aux définitions de la table de définition de canal du client.

Tâches associées

«Configuration d'une table de définition de canal du client au format binaire», à la page 45

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

Référence associée

MQI Client : Table de définition de canal du client (CCDT)

ALW

Programmes d'exit de canal pour les canaux MQI

Trois types d'exit de canal sont disponibles pour l'environnement IBM MQ MQI client sur AIX, Linux, and Windows.

Il s'agit des fonctions suivantes :

- Exit d'émission
- Exit de réception
- Exit de sécurité

Ces exits sont disponibles à la fois à l'extrémité client et à l'extrémité serveur du canal. Les exits ne sont pas disponibles pour votre application si vous utilisez la variable d'environnement **MQSERVER**. Les exits de canal sont décrits dans Programmes d'exit de canal pour les canaux de messagerie.

Les exits d'envoi et de réception fonctionnent ensemble. Vous pouvez les utiliser de plusieurs manières:

- Fractionnement et réassemblage d'un message
- Compression et décompression des données dans un message (cette fonctionnalité est fournie avec IBM MQ, mais vous pouvez utiliser une autre technique de compression)
- Chiffrement et déchiffrement des données utilisateur (cette fonctionnalité est fournie avec IBM MQ, mais vous pouvez utiliser une autre technique de chiffrement)
- Journalisation de chaque message envoyé et reçu

Vous pouvez utiliser l'exit de sécurité pour vous assurer que le client et le serveur IBM MQ sont correctement identifiés et pour contrôler l'accès.

Si des exits d'émission ou de réception côté serveur de l'instance de canal doivent effectuer des appels MQI sur la connexion à laquelle ils sont associés, ils utilisent le descripteur de connexion fourni dans la zone **MQCXP Hconn** . Vous devez savoir que les exits d'envoi et de réception de connexion client ne peuvent pas effectuer d'appels MQI.

Concepts associés

«Exits de sécurité sur une connexion client», à la page 63

Vous pouvez utiliser des programmes d'exit de sécurité pour vérifier que le partenaire à l'autre extrémité d'un canal est authentique. Des considérations spéciales s'appliquent lorsqu'un exit de sécurité est appliqué à une connexion client.

Exits utilisateur, exits API et services optionnels d'IBM MQ

Tâches associées

Extension des fonctions du gestionnaire de files d'attente

Référence associée

«Chemin d'accès aux exits», à la page 63

Un chemin par défaut pour l'emplacement des exits de canal est défini dans le fichier de configuration du client. Les exits de canal sont chargés lorsqu'un canal est initialisé.

«Identification de l'appel API dans un programme d'exit d'envoi ou de réception», à la page 65

Lorsque vous utilisez des canaux MQI pour des clients, l'octet 10 de la mémoire tampon de l'agent identifie l'appel API en cours d'utilisation lorsqu'un exit d'émission ou de réception est appelé. Cela est utile pour identifier les flux de canal qui incluent des données utilisateur et qui peuvent nécessiter un traitement tel que le chiffrement ou la signature numérique.

ALW Chemin d'accès aux exits

Un chemin par défaut pour l'emplacement des exits de canal est défini dans le fichier de configuration du client. Les exits de canal sont chargés lorsqu'un canal est initialisé.

Sur les systèmes AIX, Linux, and Windows , un fichier de configuration client est ajouté à votre système lors de l'installation de IBM MQ MQI client. Un chemin par défaut pour l'emplacement des exits de canal sur le client est défini dans ce fichier, à l'aide de la strophe:

```
ClientExitPath:  
ExitsDefaultPath= string  
ExitsDefaultPath64= string
```

où *chaîne* est un emplacement de fichier dans un format approprié à la plateforme

Lorsqu'un canal est initialisé, après un appel MQCONN ou MQCONNX , la recherche est effectuée dans le fichier de configuration du client. La section de chemin ClientExitest lue et tous les exits de canal spécifiés dans la définition de canal sont chargés.

ALW Exits de sécurité sur une connexion client

Vous pouvez utiliser des programmes d'exit de sécurité pour vérifier que le partenaire à l'autre extrémité d'un canal est authentique. Des considérations spéciales s'appliquent lorsqu'un exit de sécurité est appliqué à une connexion client.

La Figure 4, à la page 64 illustre l'utilisation des exits de sécurité dans une connexion client, à l'aide du gestionnaire des droits d'accès aux objets IBM MQ pour authentifier un utilisateur.

La zone SecurityParmsPtr ou SecurityParmsOffset de la structure MQCNO est définie par le client et il existe des exits de sécurité aux deux extrémités du canal. Une fois l'échange de messages de sécurité normal terminé et que le canal est prêt à s'exécuter, la structure MQCSP est transmise à l'exit de sécurité du client. L'exit peut accéder à la structure MQCSP à l'aide de la zone SecurityParms de la structure MQCXP. Le type d'exit est défini sur MQXR_SEC_PARMS. L'exit de sécurité peut modifier les données d'identification dans la structure MQCSP ou les laisser inchangées.

Les données renvoyées par l'exit sont ensuite envoyées à l'extrémité de connexion serveur du canal. La structure MQCSP est régénérée à l'extrémité de connexion serveur du canal et est transmise à l'exit de sécurité de connexion serveur. L'exit peut accéder à la structure MQCSP à l'aide de la zone SecurityParms de la structure MQCXP. L'exit de sécurité reçoit et traite ces données. Ce traitement permet généralement d'annuler toute modification apportée aux données d'identification dans la structure MQCSP par l'exit client, qui est ensuite utilisé pour autoriser la connexion du gestionnaire de files d'attente. La structure MQCSP résultante est référencée à l'aide de SecurityParmsPtr dans la structure MQCNO sur le système du gestionnaire de files d'attente.

L'adresse mémoire renvoyée avec la zone SecurityParms de la structure MQCXP doit rester adressable et inchangée jusqu'à MQXR_TERM. Un exit ne doit pas invalider ou libérer la mémoire sur le système avant que l'exit ne soit appelé pour MQXR_TERM.

Si la zone SecurityParmsPtr ou SecurityParmsOffset de la structure MQCNO est définie et qu'il existe un exit de sécurité à une seule extrémité du canal, l'exit de sécurité reçoit et traite la structure MQCSP. Les actions telles que le chiffrement sont inappropriées pour un exit utilisateur unique, car il n'y a pas d'exit pour effectuer l'action complémentaire.

Si les zones SecurityParmsPtr et SecurityParmsOffset de la structure MQCNO ne sont pas définies et qu'il existe un exit de sécurité à l'une des extrémités du canal ou aux deux extrémités du canal, l'exit ou les exits de sécurité sont appelés. Chaque exit de sécurité peut renvoyer sa propre structure

MQCSP qui est traitée par la zone SecurityParmsPtr . L'exit de sécurité n'est pas appelé à nouveau tant qu'il n'est pas arrêté (ExitReason de MQXR_TERM). Le programme d'écriture d'exit peut libérer la mémoire utilisée pour le MQCSP à ce stade.

Lorsqu'une instance de canal de connexion serveur partage plusieurs conversations, le schéma des appels à l'exit de sécurité est restreint sur la deuxième conversation et les conversations suivantes.

Pour la première conversation, le modèle est le même que si l'instance de canal ne partage pas de conversations. Pour la deuxième conversation et les suivantes, l'exit de sécurité n'est jamais appelé avec MQXR_INIT, MQXR_INIT_SEC ou MQXR_SEC_MSG. Il est appelé avec MQXR_SEC_PARMS.

Dans une instance de canal avec partage de conversations, MQXR_TERM est appelé uniquement pour la dernière conversation en cours d'exécution.

Chaque conversation a la possibilité dans l'appel MQXR_SEC_PARMS de l'exit de modifier le MQCD ; sur l'extrémité de connexion serveur du canal, cette fonction peut être utile pour varier, par exemple, les valeurs MCAUserIdentifier ou LongMCAUserIdPtr avant la connexion au gestionnaire de files d'attente.

Server-connection exit	Client-connection exit
	Invoked with MQXR_INIT Responds with MQXCC_OK
Invoked with MQXR_INIT Responds with MQXCC_OK	
	Invoked with MQXR_INIT_SEC Responds with MQXCC_OK
Invoked with MQXR_INIT_SEC Responds with MQXCC_OK	
	Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK
Invoked with MQXR_SEC_PARMS Responds with MQXCC_OK	
Data transfer begins	
Invoked with MQXR_TERM Responds with MQXCC_OK	Invoked with MQXR_TERM Responds with MQXCC_OK

Figure 4. Echange initié par la connexion client avec accord pour la connexion client à l'aide de paramètres de sécurité

Remarque : Les applications d'exit de sécurité construites avant l'édition de IBM WebSphere MQ 7.1 peuvent nécessiter une mise à jour. Pour plus d'informations, voir [Programmes d'exit de sécurité de canal](#).

ALW Identification de l'appel API dans un programme d'exit d'envoi ou de réception

Lorsque vous utilisez des canaux MQI pour des clients, l'octet 10 de la mémoire tampon de l'agent identifie l'appel API en cours d'utilisation lorsqu'un exit d'émission ou de réception est appelé. Cela est utile pour identifier les flux de canal qui incluent des données utilisateur et qui peuvent nécessiter un traitement tel que le chiffrement ou la signature numérique.

Le tableau suivant présente les données qui apparaissent dans l'octet 10 du flux de canal lorsqu'un appel API est en cours de traitement.

Remarque : Ce ne sont pas les seules valeurs de cet octet. Il existe d'autres valeurs **réservées** .

<i>Tableau 8. Identification des appels d'API</i>		
appel API	Valeur de l'octet 10 pour la demande	Valeur de l'octet 10 pour la réponse
MQCONN «1», à la page 66, «2», à la page 66	X'81'	X' 91'
MQDISC «1», à la page 66	X'82'	X' 92'
MQOPEN «3», à la page 66	X'83'	X' 93'
MQCLOSE	X'84'	X' 94'
MQGET «4», à la page 66	X'85'	X' 95'
MQPUT «4», à la page 66	X'86'	X' 96'
MQPUT1 requête «4», à la page 66	X'87'	X' 97'
Demande MQSET	X'88'	X' 98'
Demande MQINQ	X'89'	X' 99'
Demande MQCMIT	X'8A'	X'9A'
MQBACK, demande	X'8B'	X'9B'
Demande MQSTAT	X'8D'	X'9D'
MQSUB, demande	X'8E'	X'9E'
MQSUBRQ, demande	X'8F'	X'9F'
Demande xa_start	X'A1'	X'B1'
Demande xa_end	X'A2'	X'B2'
xa_open, demande	X'A3'	X'B3'
Demande xa_close	X'A4'	X'B4'
xa_prepare, demande	X'A5'	X'B5'
Demande xa_commit	X'A6'	X'B6'
Demande xa_rollback	X'A7'	X'B7'
Demande xa_forget	X'A8'	X'B8'
Demande xa_recover	X'A9'	X'B9'
Demande xa_complete	X'AA'	X'BA'

Remarques :

1. La connexion entre le client et le serveur est initiée par l'application client à l'aide de MQCONN. Par conséquent, pour cette commande en particulier, il existe plusieurs autres flux de réseau. Il en va de même pour MQDISC, qui met fin à la connexion réseau.
2. MQCONNX est traité de la même manière que MQCONN pour les besoins de la connexion client-serveur.
3. Si une grande liste de distribution est ouverte, il peut y avoir plusieurs flux réseau par appel MQOPEN afin de transmettre toutes les données requises à l'agent MCA SVRCONN.
4. Les messages volumineux peuvent dépasser la taille du segment de transmission. Si cela se produit, il peut y avoir de nombreux flux de réseau résultant d'un seul appel API.

z/OS

Connecting a client to a queue sharing group

You can connect a client to a queue sharing group by creating an MQI channel between a client and a queue manager on a server that is a member of a queue sharing group.

About this task

A queue sharing group is formed by a set of queue managers that can access the same set of shared queues. For more information about shared queues, see [Shared queues and queue sharing groups](#).

A client putting to a shared queue can connect to any member of the queue sharing group. The benefits of connecting to a queue sharing group are possible increases in front-end and back-end availability, and increased capacity. You can connect to a specific queue manager or to the generic interface.

Connecting directly to a queue manager in a queue sharing group gives the benefit that you can put messages to a shared target queue, which increases back-end availability.

Connecting to the generic interface of a queue sharing group opens a session with one of the queue managers in the group. This increases front-end availability, because the client queue manager can connect with any queue manager in the group. You connect to the group using the generic interface when you do not want to connect to a specific queue manager within the queue sharing group.

The generic interface can be a Sysplex Distributor VIPA address or a VTAM generic resource name, or another common interface to the queue sharing group. For more details on setting up a generic interface, see [Setting up communication for IBM MQ for z/OS using queue sharing groups](#).

Procedure

To connect to the generic interface of a queue sharing group you need to create channel definitions that can be accessed by any queue manager in the group. To do this you must have the same definitions on each queue manager in the group.

1. Define the SVRCONN channel as shown in the following example:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
QSGDISP(GROUP)
```

Channel definitions on the server are stored in a shared Db2® repository. Each queue manager in the queue sharing group makes a local copy of the definition, ensuring that you will always connect to the correct server-connection channel when you issue an MQCONN or MQCONNX call.

2. Define the CLNTCONN channel as shown in the following example:

```
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME( VIPA address ) QMNAME(QSG1) +
DESCR('Client-connection to Queue Sharing Group QSG1') QSGDISP(GROUP)
```

Results

Because the generic interface of the queue sharing group is stored in the CONNAME field in the client-connection channel, you can now connect to any queue manager in the group, and put to shared queues owned by that group.

Utilisation des variables d'environnement IBM MQ

Vous pouvez utiliser des commandes pour afficher les paramètres en cours ou pour réinitialiser les valeurs des variables d'environnement IBM MQ .

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser les variables d'environnement comme suit:

- Pour définir les variables dans votre profil système afin d'effectuer une modification permanente
- Pour émettre une commande à partir de la ligne de commande afin d'effectuer une modification pour cette session uniquement
- Pour attribuer à une ou plusieurs variables une valeur particulière en fonction de l'application en cours d'exécution, ajoutez des commandes à un fichier script de commandes utilisé par l'application

Pour chaque variable d'environnement, vous pouvez utiliser des commandes pour afficher le paramètre en cours ou pour réinitialiser la valeur de la variable d'environnement. Ces commandes sont disponibles sur toutes les plateformes prises en charge, sauf indication contraire. Le format de la commande dépend de votre plateforme. Exemple :

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

```
export [environment variable]=value
```

- **Windows** Sous Windows :

```
Set [environment variable]=value
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(environment variable) VALUE(xx)
```

- **MQ Appliance** Pour IBM MQ Appliance, voir [Configuration des variables d'environnement sur IBM MQ Appliance](#) dans la documentation IBM MQ Appliance .

Le cas échéant, IBM MQ utilise des valeurs par défaut pour les variables d'environnement que vous n'avez pas définies.

Remarque : **z/OS** IBM MQ for z/OS ne prend en charge aucune variable d'environnement IBM MQ . Si vous utilisez cette plateforme comme serveur, voir [Table de définition de canal du client](#) pour plus d'informations sur la façon dont la table de définition de canal du client est générée sur z/OS. Vous pouvez toujours utiliser les variables d'environnement IBM MQ sur votre plateforme client.

Procédure

- **Windows**

Sous Windows, pour chaque variable d'environnement, utilisez les commandes suivantes pour afficher le paramètre en cours ou pour réinitialiser la valeur d'une variable:

- Pour supprimer la valeur d'une variable d'environnement, utilisez la commande suivante:

```
SET MQSERVER=
```

- Pour afficher le paramètre en cours d'une variable d'environnement, utilisez la commande suivante:

```
SET MQSERVER
```

- Pour afficher toutes les variables d'environnement de la session, utilisez la commande suivante:

```
set
```

- **Linux** **AIX**

Sous AIX and Linux, pour chaque variable d'environnement, utilisez les commandes suivantes pour afficher le paramètre en cours ou pour réinitialiser la valeur d'une variable:

- Pour supprimer la valeur d'une variable d'environnement, utilisez la commande suivante:

```
unset MQSERVER
```

- Pour afficher le paramètre en cours d'une variable d'environnement, utilisez la commande suivante:

```
echo $MQSERVER
```

- Pour afficher toutes les variables d'environnement de la session, utilisez la commande suivante:

```
set
```

Tâches associées

[Définition des variables d'environnement pour IBM MQ classes for JMS/Jakarta Messaging](#)

[Variables d'environnement relatives à IBM MQ classes for Java](#)

[Définition de variables d'environnement supplémentaires dans le fichier service.env](#)

[«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms», à la page 96](#)

Vous pouvez modifier le comportement d' IBM MQ ou d'un gestionnaire de files d'attente individuel en fonction des besoins de votre installation en éditant les informations dans les fichiers de configuration (.ini). Vous pouvez également modifier les options de configuration pour IBM MQ MQI clients.

Référence associée

[Utilisation des variables d'environnement dans les propriétés MFT](#)

Description des variables d'environnement

Description des variables d'environnement serveur et client destinées à être utilisées par le client.

Exemples d'utilisation

- **Linux** **AIX** Sur les systèmes AIX and Linux , utilisez le format suivant: `export [environment variable]=value`.
- **Windows** Sur les systèmes Windows , utilisez le format suivant: `Set [environment variable]=value`.
- **IBM i** Sur les systèmes IBM i , utilisez le format suivant: `ADDENVVAR ENVVAR(environment variable) VALUE(xx)`.
- **MQ Appliance** Pour IBM MQ Appliance, voir [Configuration des variables d'environnement sur IBM MQ Appliance](#) dans la documentation IBM MQ Appliance .

Tableau 9. Récapitulatif des variables d'environnement

Nom	Description	Sur le serveur?	Sur le client?
AMQ_ALLOWED_CIPHERS	Indique une liste personnalisée de CipherSpecs qui sont activés pour être utilisés avec les canaux IBM MQ .	✓	

Tableau 9. Récapitulatif des variables d'environnement (suite)

Nom	Description	Sur le serveur?	Sur le client?
<u>AMQ_BAD_COMMS_DATA_FDCS</u>	Entraîne l'écriture des fichiers FFST pour les données incorrectes, y compris les formats connus simples, si IBM MQ reçoit des données dans un format incorrect d'un hôte sur TCP/IP.	✓	
<u>AMQ_CONVEBCDICNEWLINE</u>	Indique comment IBM MQ doit convertir un caractère NL EBCDIC au format ASCII.	✓	
<u>AMQ_DIAGNOSTIC_MESSAGE_SEVERITY</u>	Indique si la gravité du message doit être ajoutée au numéro de message lorsqu'un processus IBM MQ écrit un message dans un journal des erreurs ou dans la console.	✓	
<u>AMQ_DISABLE_CLIENT_AMS</u>	Pour les clients Java , si vous souhaitez vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit, désactivez IBM MQ Advanced Message Security (AMS) sur le client.		✓
<u>AMQ_DMPMQCFG_QSGDISP_DEFAULT</u>	Indique les types de définition à inclure lors de l'interrogation de la disposition d'un gestionnaire de files d'attente.	✓	
V9.4.0 <u>AMQ_IODELAY</u> , <u>AMQ_IODELAY_INMS</u> et <u>AMQ_IODELAY_FFST</u>	Permet d'optimiser les diagnostics et les délais lorsque l'entrée et la sortie du journal de votre gestionnaire de files d'attente et du système de fichiers de stockage sont lentes.	✓	
<u>AMQ_LDAP_TRACE</u>	Permet d'activer et de désactiver la trace du client LDAP sans arrêter ou démarrer le gestionnaire de files d'attente.	✓	
<u>AMQ_LICENSING_METRIC</u>	Fait en sorte que le gestionnaire de files d'attente télécharge les données relatives aux types de licence VPC mensuels au lieu du comportement par défaut de téléchargement des données relatives aux licences à base de conteneurs horaires.	✓	
Linux AIX <u>AMQ_MQS_INI_LOCATION</u>	Indique l'emplacement utilisé pour le fichier mqs.ini.	✓	
<u>AMQ_NO_BAD_COMM_DATA_FDCS</u>	Supprime la génération de FFST lors de la génération de messages d'erreur sur le flux de communications initial si les données que IBM MQ reçoit d'un hôte sur TCP/IP sont dans un format incorrect.	✓	
<u>AMQ_NO_IPV6</u>	Désactive l'utilisation de IPv6 lors de la tentative de connexion.	✓	✓

Tableau 9. Récapitulatif des variables d'environnement (suite)







Nom	Description	Sur le serveur?	Sur le client?
<u>AMQ_REVERSE_COMM</u> <u>IT_ORDER</u>	Configure un gestionnaire de files d'attente de sorte que, dans une transaction XA, la modification du gestionnaire de files d'attente IBM MQ soit validée après la mise à jour de la base de données correspondante.	✓	
<u>AMQ_SSL_ALLOW_DE</u> <u>FAULT_CERT</u>	Permet au certificat utilisé par une application pour se connecter à un gestionnaire de files d'attente d'être un certificat par défaut, à condition qu'un certificat par défaut soit présent dans le référentiel de clés et que le référentiel de clés ne contienne pas de certificat personnel avec le préfixe <code>ibmwebsphereuserid</code> .	✓	
<u>AMQ_SSL_LDAP_SERV</u> <u>ER_VERSION</u>	Indique que LDAP v2 ou LDAP v3 est utilisé par les composants cryptographiques IBM MQ dans les cas où les serveurs CRL requièrent l'utilisation d'une version spécifique du protocole LDAP.	✓	
 <u>AMQ_USE_ZLIBNX</u>	Sous AIX, permet aux agents MCA d'utiliser la bibliothèque <code>zlibNX</code> à accélération matérielle pour la compression et la décompression des données de message lors de l'utilisation des techniques <code>ZLIBFAST</code> ou <code>ZLIBHIGH</code> .	✓	
<u>GMQ_MQ_LIB</u>	Indique la bibliothèque de liaisons client lorsque les classes d'automatisation IBM MQ pour ActiveX (MQAX) sont exécutées sur le client et non sur le serveur.  Avertissement : Cette variable d'environnement est supprimée dans la version 9.2.		
 <u>ACCUEIL</u>  	Sous AIX, Linux et IBM i, indique le nom du répertoire dans lequel le fichier <code>mqclient.ini</code> est recherché.	✓	
 <u>HOMEDRIVE</u> et <u>HOMEPATH</u>	Sous Windows, utilisé ensemble pour spécifier le nom du répertoire dans lequel le fichier <code>mqclient.ini</code> est recherché.	✓	
<u>LDAP_BASEDN</u>	Environnement requis pour l'exécution d'un exemple de programme LDAP. Indique le nom distinctif de base pour la recherche dans l'annuaire.	✓	

Tableau 9. Récapitulatif des variables d'environnement (suite)


Nom	Description	Sur le serveur?	Sur le client?
<u>HOTE_LDAP</u>	Facultatif pour l'exécution d'un exemple de programme LDAP. Indique le nom de l'hôte sur lequel le serveur LDAP s'exécute.	✓	
<u>VERSION_LDAP</u>	Facultatif pour l'exécution d'un exemple de programme LDAP. Indique la version du protocole LDAP à utiliser.	✓	
<u>MQ_CHANNEL_SUPPRESS_INTERVAL</u>	Indique l'intervalle de temps, en secondes, pendant lequel les messages définis avec MQ_CHANNEL_SUPPRESS_MSGS doivent être supprimés de l'écriture dans le journal des erreurs, ainsi que le nombre de fois où un message sera autorisé à se produire pendant l'intervalle de temps spécifié avant d'être supprimé.	✓	
<u>MQ_CHANNEL_SUPPRESS_MSGS</u>	Supprime les messages d'erreur de canal dans le journal des erreurs.	✓	
<u>MQ_CONNECT_TYPE</u>	Utilisé en combinaison avec le type de liaison spécifié dans la zone Options de la structure MQCNO utilisée sur un appel MQCONN.		✓
<u>MQ_CROSS_QUEUE_ORDER_ALL</u>	Indique que l'ordre d'insertion de message est géré dans une unité d'oeuvre.	✓	
<u>MQ_PREFIX_EPHEMERAL</u>	Indique le chemin d'accès au répertoire éphémère du gestionnaire de files d'attente, dans lequel les données du gestionnaire de files d'attente éphémères sont conservées, alors que le gestionnaire de files d'attente est en cours d'exécution.	✓	
 <u>CHEMIN_FILE_MQ</u>	Créé lors de l'installation du package d'exécution. Contient les mêmes données que le registre Windows .	✓	
<u>MQ_JAVA_DATA_PATH</u>	Indique le répertoire de la sortie de journal et de trace pour IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging, ainsi que pour IBM MQ classes for Java.	✓	
<u>MQ_JAVA_INSTALL_PATH</u>	Indique le répertoire dans lequel IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging, ainsi que IBM MQ classes for Java , sont installés.	✓	

Tableau 9. Récapitulatif des variables d'environnement (suite)



Nom	Description	Sur le serveur?	Sur le client?
<u>MQ_JAVA_LIB_PATH</u>	Indique le répertoire dans lequel sont stockées les bibliothèques IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging, ainsi que les bibliothèques IBM MQ classes for Java .	✓	
«CHEMIN_DATA_OVERRIDE_MQ», à la page 82	Remplace le répertoire par défaut du chemin de données IBM MQ .	✓	✓
 <u>MQ_SET_NODELAYACK</u>	Sous AIX, désactive l'accusé de réception retardé TCP.	✓	
«NOM_UTILISATEUR_MQ», à la page 82	Permet à une installation non enregistrée sur Linux de choisir le nom d'un utilisateur sans nom.	✓	
<u>MQAPI_TRACE_LOGFILE</u>	Définit le préfixe du fichier spécifié par l'utilisateur dans lequel l'exemple de programme d'exit API génère une trace MQI.	✓	
 <u>MQAPPLNAME</u>	Si le nom de l'application n'a pas encore été choisi, indique le nom à utiliser pour identifier la connexion à un gestionnaire de files d'attente.		✓
<u>MQCCSID</u>	Indique le numéro de jeu de caractères codés à utiliser et remplace la valeur CCSID avec laquelle le serveur a été configuré.		✓
<u>MQCCDTURL</u>	Offre la capacité équivalente à la définition d'une combinaison des variables d'environnement MQCHLLIB et MQCHLTAB .		✓
<u>MQCERTLABL</u>	Définit le libellé de certificat d'une définition de canal à utiliser par IBM MQ pour localiser un certificat personnel envoyé lors de l'établissement d'une liaison TLS.		✓
<u>MQCERTVPOL</u>	Indique le type de règle de validation de certificat à utiliser.		✓
<u>MQCHLLIB</u>	Indique le chemin de répertoire du fichier contenant la table de définition de canal du client (CCDT).		✓
<u>MQCHLTAB</u>	Indique le nom du fichier contenant la table de définition de canal du client (CCDT).		✓
<u>MQCLNTCF</u>	Spécifie l'emplacement du fichier de configuration IBM MQ MQI client.		✓

Tableau 9. Récapitulatif des variables d'environnement (suite)



Nom	Description	Sur le serveur?	Sur le client?
<u>MQDOTNET_TRACE_ON</u>	Active la trace pour les clients redistribuables IBM MQ .NET .		✓
<u>MQIPADDRV</u>	Indique le protocole IP à utiliser pour une connexion de canal.		✓
<u>MQKEYRPWD</u>	Indique le mot de passe du référentiel de clés qui contient le certificat numérique appartenant à l'utilisateur.		✓
 <u>MQLICENSE</u>	Sur les systèmes Linux , utilisé pour accepter ou afficher une licence IBM MQ après l'installation du produit.	✓	✓
<u>MQMAXERRORLOGSIZE</u>	Indique la taille du journal des erreurs du gestionnaire de files d'attente qui est copié dans la sauvegarde.	✓	
 <u>NOMMQ</u>	Indique le nom NetBIOS local que les processus IBM MQ peuvent utiliser.	✓	✓
<u>MQNOREMPOOL</u>	Désactive le regroupement de canaux et entraîne l'exécution des canaux en tant qu'unités d'exécution du programme d'écoute.	✓	
<u>MQPSE_TRACE_LOGFILE</u>	Décrit l'emplacement dans lequel les fichiers de trace du programme exemple d'exit de publication doivent être écrits.	✓	
<u>MQS_AMSCRED_KEYFILE</u>	Remplace ou fournit le fichier de clés initial à utiliser lors de l'exécution des applications IBM MQ Advanced Message Security (AMS) ou lorsque vous protégez un fichier de configuration de magasin de clés à l'aide de la commande runamscred .		✓
<u>MQS_DISABLE_ALL_INTERCEPT</u>	Pour les clients C natifs, si vous souhaitez vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit, désactivez IBM MQ Advanced Message Security (AMS) sur le client.		✓
<u>MQS_IPC_HOTE</u>	Définit le nom d'hôte qui est ajouté au chemin de répertoire.	✓	
<u>MQS_KEYSTORE_CONFIG</u>	Indique l'emplacement du fichier de configuration du magasin de clés pour IBM MQ Advanced Message Security (AMS), si le fichier ne se trouve pas à l'emplacement par défaut.		✓
<u>MQS_MQI_KEYFILE</u>	Indique l'emplacement d'un fichier de clés initial qui contient la clé initiale à utiliser pour les opérations de protection par mot de passe.		✓

Tableau 9. Récapitulatif des variables d'environnement (suite)



Nom	Description	Sur le serveur?	Sur le client?
<u>MQS_SSLCRYP_KEYFILE</u>	Indique le chemin d'accès complet et le nom du fichier contenant la clé initiale utilisée pour chiffrer le mot de passe dans la chaîne de configuration du matériel de cryptographie PKCS #11 .		✓
<u>OPTIONS-TRACE_MQ</u>	Pour le traçage sélectif des composants sous AIX, active les fonctions de traçage détaillé et de paramètre de manière individuelle.	✓	✓
<u>MQSERVER</u>	Définit un canal minimal, en spécifiant l'emplacement du serveur IBM MQ et la méthode de communication à utiliser.		✓
<u>MQSNOAUT</u>	Désactive le gestionnaire des droits d'accès aux objets (OAM) et empêche toute vérification de la sécurité, par exemple dans un environnement de test.	✓	
<u>MQSPREFIX</u>	Au lieu de modifier le préfixe par défaut, remplace le préfixe DefaultPrefix pour la commande crtmqm .	✓	
 <u>MQSSLCRYP</u>	Contient une chaîne de paramètres que vous pouvez utiliser pour configurer le matériel cryptographique présent sur le système.		✓
<u>MQSSLFIPS</u>	Indique si seuls les algorithmes certifiés FIPS doivent être utilisés si la cryptographie est effectuée dans IBM MQ.		✓
<u>MQSSLKEYR</u>	Indique l'emplacement du référentiel de clés qui contient le certificat numérique appartenant à l'utilisateur.		Client ✓ et outil mqcertck
<u>MQSSLPROXY</u>	Indique le nom d'hôte et le numéro de port du serveur proxy HTTP à utiliser par IBM Global Security Kit (GSKit) pour les vérifications OCSP.		✓
<u>MQSSLRESET</u>	Indique le nombre d'octets non chiffrés envoyés et reçus sur un canal TLS avant que la clé secrète TLS ne soit renégo-ciée.		✓
 <u>MQSUIEB</u>	Indique si la cryptographie compatible Suite B doit être utilisée. Dans le cas où la cryptographie Suite B est utilisée,		✓
<u>MQTCPTIMEOUT</u>	indique combien de temps IBM MQ attend un appel de connexion TCP.	✓	✓
<u>MSG_ODQ</u>	Définit le nom du fichier contenant les messages d'erreur et d'information, si vous utilisez un gestionnaire de files d'attente de rebut différent de runmqdlq .	✓	

Tableau 9. Récapitulatif des variables d'environnement (suite)

Nom	Description	Sur le serveur?	Sur le client?
<u>TRACE ODQ</u>	Active la fonction de trace si vous utilisez un gestionnaire de files d'attente de rebut différent de runmqdlq .	✓	
<u>WCF_TRACE_ON</u>	Active la trace pour le canal personnalisé WCF.		✓
<u>REP_BASE_WMQSO</u>	Utilisé lors de l'exécution d'étapes de configuration supplémentaires après l'installation et la configuration correctes de l'environnement d'hébergement de service .NET SOAP sur JMS dans IBM MQ.		✓
<u>XMS_TRACE_ON</u> , <u>XMS_TRACE_FILE_PATH</u> , <u>XMS_TRACE_FORMAT</u> et <u>XMS_TRACE_SPECIFICATION</u>	Permet d'activer et de configurer la trace XMS .		✓

AMQ_ALLOWED_CIPHERS



Vous pouvez utiliser la variable d'environnement **AMQ_ALLOWED_CIPHERS** pour spécifier une liste personnalisée de CipherSpecs qui sont activés pour être utilisés avec les canaux IBM MQ sur Multiplatforms. La variable d'environnement prend les mêmes valeurs que l'attribut de strophe SSL **AllowedCipherSpecs** du fichier `.ini` :

- Un nom CipherSpec unique, ou
- Liste de noms IBM MQ CipherSpec séparés par des virgules à réactiver, ou
- Valeur spéciale de ALL, représentant tous les CipherSpecs (non recommandé).

Remarque : L'activation de **ALL** CipherSpecs n'est pas recommandée car cela active les protocoles SSL 3.0 et TLS 1.0 et un grand nombre d'algorithmes de cryptographie faibles.

Pour plus d'informations, voir [Ajout d'une liste personnalisée de CipherSpecs sur Multiplatforms dans CipherSpec dans l'établissement de liaison TLS](#).

AMQ_BAD_COMMS_DATA_FDCS

La variable d'environnement **AMQ_BAD_COMMS_DATA_FDCS** est effective lorsqu'elle est définie sur n'importe quelle valeur.

Si les données que IBM MQ reçoit d'un hôte sur TCP/IP sont dans un format incorrect, par exemple parce qu'un client réseau s'est connecté à un port d'écoute IBM MQ et a tenté de communiquer avec un protocole d'application non pris en charge, le gestionnaire de files d'attente écrit un message d'erreur **AMQ9207E** dans les journaux d'erreurs du gestionnaire de files d'attente. Les programmes d'écoute IBM MQ prennent en charge les connexions TCP/IP à partir des agents MCA et des applications client MQI, JMS et XMS .

Remarque : Les programmes d'écoute IBM MQ ne prennent pas en charge le protocole d'application utilisé par les clients AMQP et MQTT. Ces clients doivent plutôt se connecter aux ports réseau configurés dans le canal AMQP ou le service de télémétrie MQXR applicable.

Un enregistrement de capture de données d'échec (FDC) contenant les données non valides reçues par IBM MQ peut également être écrit. Toutefois, un fichier FFST n'est pas généré s'il s'agit du début d'une

conversation avec le côté distant et que le format est un format connu simple tel qu'une requête GET provenant d'un navigateur Web HTTP. Si vous souhaitez remplacer cette valeur pour que les fichiers FFST soient écrits pour toutes les données incorrectes, y compris les formats connus simples, vous pouvez définir la variable d'environnement **AMQ_BAD_COMMS_DATA_FDCS** sur n'importe quelle valeur (par exemple, TRUE) et redémarrer le gestionnaire de files d'attente.

AMQ_CONVEBCDICNEWLINE



Vous pouvez utiliser la variable d'environnement **AMQ_CONVEBCDICNEWLINE** pour spécifier comment IBM MQ doit convertir un caractère NL EBCDIC au format ASCII. La variable d'environnement prend les mêmes valeurs que l'attribut **ConvEBCDICNewLine** de `mqs.ini`, c'est-à-dire `NL_TO_LF`, `TABLE` ou `ISO` (voir la section [Tous les gestionnaires de files d'attente du fichier mqs.ini](#)). Vous pouvez, par exemple, utiliser la variable d'environnement **AMQ_CONVEBCDICNEWLINE** à la place de l'attribut de section **ConvEBCDICNewLine** pour fournir la fonctionnalité **ConvEBCDICNewLine** côté client dans les situations où le fichier `mqs.ini` ne peut pas être utilisé. Si l'attribut de section et la variable d'environnement sont définis, l'attribut de section est prioritaire.

Pour plus d'informations, voir [Conversion de données entre des jeux de caractères codés](#).

AMQ_DIAGNOSTIC_MSG_SEVERITY

Si la variable d'environnement **AMQ_DIAGNOSTIC_MSG_SEVERITY** est définie sur 1 pour un processus IBM MQ, la gravité du message est ajoutée au numéro de message sous la forme d'un caractère alphabétique majuscule unique lorsque le processus IBM MQ écrit un message dans un journal des erreurs ou dans la console.

Le comportement activé par **AMQ_DIAGNOSTIC_MSG_SEVERITY** est défini par défaut. Vous pouvez désactiver ce comportement en définissant la variable d'environnement sur 0.

Pour plus d'informations, voir [Utilisation des journaux d'erreurs](#).

AMQ_DISABLE_CLIENT_AMS

Vous pouvez utiliser la variable d'environnement **AMQ_DISABLE_CLIENT_AMS** pour désactiver IBM MQ Advanced Message Security (AMS) sur le client si une erreur 2085 (`MQRX_UNKNOWN_OBJECT_NAME`) est signalée lorsque vous tentez de vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit et que vous utilisez l'un des clients suivants:

- Un environnement Java runtime environment (JRE) autre que IBM Java runtime environment (JRE)
- Un client IBM MQ IBM MQ classes for JMS ou IBM MQ classes for Java.

Remarque : Vous ne pouvez pas utiliser la variable d'environnement **AMQ_DISABLE_CLIENT_AMS** pour les clients C. Vous devez utiliser la variable d'environnement **MQS_DISABLE_ALL_INTERCEPT** à la place.

Pour plus d'informations, voir [Désactivation d'Advanced Message Security sur le client](#).

AMQ_DMPMQCFG_QSGDISP_DEFAULT

Les interrogations sur la disposition d'un gestionnaire de files d'attente qui sont utilisées par la commande **dmpmqcfg** interrogent uniquement les définitions QSGDISP (QMGR) par défaut. Vous pouvez consulter des définitions supplémentaires à l'aide de la variable d'environnement **AMQ_DMPMQCFG_QSGDISP_DEFAULT**, qui peut être définie sur l'une des valeurs suivantes:

lancé

Incluez uniquement les objets définis avec QSGDISP (QMGR) ou QSGDISP (COPY).

TOUT

Inclure les objets définis avec QSGDISP (QMGR) et QSGDISP (COPY). Si le gestionnaire de files d'attente est membre d'un groupe de partage de files d'attente, QSGDISP (GROUP) et QSGDISP (SHARED) sont également inclus.

COPIER

Inclusion uniquement, objets définis avec QSGDISP (COPY)

GROUPE

Inclure uniquement les objets définis avec QSGDISP (GROUP) ; le gestionnaire de files d'attente cible doit être membre d'un groupe de partage de files d'attente.

QMGR

Inclure uniquement les objets définis avec QSGDISP (QMGR). Il s'agit du comportement par défaut si vous utilisez cette variable d'environnement pour correspondre au comportement existant de **dmpmqcfig**.

PRIVATE

Incluez uniquement les objets définis avec QSGDISP (QMGR) ou QSGDISP (COPY).

PARTAGÉ

Inclure uniquement les objets définis avec QSGDISP (SHARED).

AMQ_IODELAY, AMQ_IODELAY_INMS et AMQ_IODELAY_FFST



IBM MQ détecte lorsque les opérations de lecture et d'écriture de journal ou les opérations d'entrée et de sortie prennent plus de temps que prévu. Cela peut être dû à des problèmes liés au système d'exploitation ou au système de stockage et peut avoir un impact sur les performances du gestionnaire de files d'attente. Depuis la IBM MQ 9.4.0, vous pouvez utiliser les variables d'environnement **AMQ_IODELAY** pour optimiser les diagnostics et les délais lorsque l'entrée et la sortie du journal du gestionnaire de files d'attente et du système de fichiers de stockage sont lentes. Si le message `AMQ6729W Log I/O operation exceeded threshold` apparaît dans le journal des erreurs du gestionnaire de files d'attente, examinez la cause et effectuez les ajustements nécessaires. Utilisez les variables comme indiqué dans les exemples suivants:

AMQ_IODELAY

Temps de seuil en secondes, la valeur par défaut est 1 seconde. Si une opération d'E-S dure plus longtemps que ce seuil, le message d'erreur AMQ6729W est consigné dans les fichiers journaux IBM MQ. Le message d'avertissement est répété au maximum toutes les 10 secondes si les délais se poursuivent. Vous pouvez augmenter cette valeur pour supprimer les erreurs ou la réduire afin d'examiner des problèmes de performances spécifiques. Exemple :

```
export AMQ_IODELAY=200000
```

AMQ_IODELAY_INMS

Changez la mesure du temps en microsecondes au lieu de secondes. Utilisez cette option pour définir un seuil inférieur avant d'obtenir le message AMQ6729 dans le journal du gestionnaire de files d'attente.

```
export AMQ_IODELAY_INMS=YES
```

AMQ_IODELAY_FFST

Outre le message d'avertissement du journal des erreurs, un fichier FFST contenant des informations de diagnostic est généré chaque fois que le seuil est dépassé.

```
export AMQ_IODELAY_FFST=YES
```

Le démarrage du gestionnaire de files d'attente comme dans cet exemple entraîne l'écriture d'un fichier FDC ou FFST si une opération d'entrée-sortie prend plus de 200000 microsecondes (0.2s), ce qui reste un seuil relativement généreux.

Pour plus d'informations, voir [Comportement du diagnostic d'intégrité du gestionnaire de files d'attente](#).

AMQ_LDAP_TRACE

Si la variable d'environnement **AMQ_LDAP_TRACE** est définie sur une valeur non nulle, il est possible d'activer et de désactiver la trace du client LDAP sans également arrêter ou démarrer le gestionnaire de files d'attente.

Pour plus d'informations, voir [Enabling dynamic tracing of LDAP client library code](#).

MÉMOIRE_CONCÉDANT_AMQ

Multi

La définition de la variable d'environnement **AMQ_LICENSING_METRIC=VPCMonthlyPeak** permet au gestionnaire de files d'attente de télécharger des données relatives aux types de licence VPC mensuels, au lieu du comportement par défaut de téléchargement des données relatives aux licences de conteneur horaires.

Pour plus d'informations sur la configuration de IBM MQ en vue de son utilisation avec le service de décompte IBM Cloud Private, voir [IBM Cloud Private metering service](#) dans la documentation IBM Cloud Private.

AMQ_MQS_INI_LOCATION

Linux AIX

Sur les systèmes AIX and Linux, vous pouvez modifier l'emplacement utilisé pour le fichier `mqs.ini` en définissant l'emplacement du fichier `mqs.ini` dans la variable d'environnement **AMQ_MQS_INI_LOCATION**. Cette variable d'environnement doit être définie au niveau du système.

Pour plus d'informations sur le fichier `mqs.ini`, y compris les emplacements de répertoire, voir le fichier de configuration [IBM MQ](#), `mqs.ini`.

AMQ_NO_BAD_COMMS_DATA_FDCS

La variable d'environnement **AMQ_NO_BAD_COMMS_DATA_FDCS** est effective lorsqu'elle est définie sur n'importe quelle valeur.

Si IBM MQ ne reconnaît pas la transmission de données initiale lors de la tentative de connexion d'un client non-IBM MQ à un programme d'écoute TCP/IP IBM MQ, le gestionnaire de files d'attente écrit un message d'erreur [AMQ9207E](#) dans les journaux d'erreurs du gestionnaire de files d'attente. Un enregistrement de capture de données de panne (FDC) est également écrit. Vous pouvez supprimer la génération de ces fichiers de diagnostic à l'aide de la variable d'environnement **AMQ_NO_BAD_COMMS_DATA_FDCS**. Lorsque **AMQ_NO_BAD_COMMS_DATA_FDCS** est défini sur une valeur (par exemple, TRUE), cela indique à IBM MQ de ne pas générer de FFST lors de la génération de messages d'erreur [AMQ9207E](#) sur le flux de communications initial. Pour être efficace, la variable d'environnement doit être définie avant le démarrage des processus du gestionnaire de files d'attente et du programme d'écoute.

Le FDC continue d'être généré dans le cas où un client envoie des flux de protocole IBM MQ valides au gestionnaire de files d'attente, puis envoie des données non valides, car cela indique un problème client qui mérite d'être examiné plus en détail.

Remarque : La capture des FFST lors de la génération de messages d'erreur [AMQ9207E](#) sur les flux de communications initiaux est supprimée par défaut.

AMQ_NO_IPV6

La variable d'environnement **AMQ_NO_IPV6** est effective lorsqu'elle est définie sur n'importe quelle valeur. Lorsque cette variable d'environnement est définie, elle désactive l'utilisation de IPv6 lors de la tentative de connexion.

AMQ_REVERSE_COMMIT_ORDER

La variable d'environnement **AMQ_REVERSE_COMMIT_ORDER** configure un gestionnaire de files d'attente de sorte que, dans une transaction XA, la modification du gestionnaire de files d'attente IBM MQ soit validée une fois la mise à jour de la base de données correspondante terminée. Les applications qui lisent les messages à partir des files d'attente voient un message uniquement une fois la mise à jour de la base de données correspondante terminée.

Remarque : Ne définissez pas **AMQ_REVERSE_COMMIT_ORDER** sans lire et comprendre le scénario décrit dans [Niveau d'isolement](#).

AMQ_SSL_ALLOW_DEFAULT_CERT

Lorsque la variable d'environnement **AMQ_SSL_ALLOW_DEFAULT_CERT** n'est pas définie, une application peut se connecter à un gestionnaire de files d'attente avec un certificat personnel dans le magasin de clés client uniquement lorsque le certificat inclut le nom de libellé `ibmwebspheremquserid`. Lorsque la variable d'environnement **AMQ_SSL_ALLOW_DEFAULT_CERT** est définie, le certificat ne requiert pas le nom de libellé `ibmwebspheremquserid`. En d'autres termes, le certificat utilisé pour la connexion à un gestionnaire de files d'attente peut être un certificat par défaut, à condition qu'un certificat par défaut soit présent dans le référentiel de clés et que le référentiel de clés ne contienne pas de certificat personnel avec le préfixe `ibmwebspheremquserid`.

La valeur 1 permet d'utiliser un certificat par défaut.

Au lieu d'utiliser la variable d'environnement **AMQ_SSL_ALLOW_DEFAULT_CERT**, une application peut utiliser le paramètre **CertificateLabel** de la strophe SSL dans le fichier `mqclient.ini`. Pour plus d'informations, voir [Labels de certificat numérique](#), [compréhension des exigences](#) et [strophe SSL](#) du fichier de configuration client.

VERSION SERVEUR_SSL_LDAP

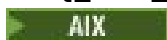
La variable d'environnement **AMQ_SSL_LDAP_SERVER_VERSION** permet de s'assurer que LDAP v2 ou LDAP v3 est utilisé par les composants cryptographiques IBM MQ lorsque les serveurs CRL requièrent l'utilisation d'une version spécifique du protocole LDAP.

Définissez la variable d'environnement sur la valeur appropriée dans l'environnement utilisé pour démarrer le gestionnaire de files d'attente ou le canal:

- Pour demander que LDAP v2 soit utilisé, définissez `AMQ_SSL_LDAP_SERVER_VERSION=2`.
- Pour demander que LDAP v3 soit utilisé, définissez `AMQ_SSL_LDAP_SERVER_VERSION=3`.

Cette variable d'environnement n'affecte pas les connexions LDAP établies par le gestionnaire de files d'attente IBM MQ pour l'authentification d'utilisateur ou l'autorisation d'utilisateur.

AMQ_USE_ZLIBNX



Sous AIX, la variable d'environnement **AMQ_USE_ZLIBNX** peut être utilisée pour permettre aux agents MCA d'utiliser la bibliothèque `zlibNX` à accélération matérielle pour la compression et la décompression des données de message lors de l'utilisation des techniques `ZLIBFAST` ou `ZLIBHIGH`.

Conseil : Les messages hautement compressibles dont la taille est supérieure à 2 Ko sont les plus susceptibles de bénéficier de l'utilisation de la bibliothèque `zlibNX`, en réduisant l'utilisation de l'unité centrale.

La bibliothèque `zlibNX` est disponible dans IBM AIX 7.2 avec Technology Level 4 Expansion Pack et versions ultérieures. Si la variable d'environnement est définie et que la bibliothèque `zlibNX` (`/usr/opt/zlibNX/lib/libz.a`) n'est pas installée, les agents MCA utiliseront la bibliothèque `zlib` standard fournie dans l'installation IBM MQ for AIX.

ACCUEIL



Sous AIX, Linux et IBM i, la variable d'environnement **HOME** spécifie le nom du répertoire dans lequel le fichier `mqclient.ini` est recherché. Ce fichier contient les informations de configuration utilisées par IBM MQ MQI clients.

Pour plus d'informations, voir [IBM MQ le fichier de configuration du client MQI, mqclient.ini et l'emplacement du fichier de configuration du client.](#)

HOMEDRIVE et HOMEPATH



Pour être utilisées, les variables d'environnement **HOMEDRIVE** et **HOMEPATH** doivent être définies. Ils sont utilisés sur les systèmes Windows pour spécifier le nom du répertoire dans lequel le fichier `mqclient.ini` est recherché. Ce fichier contient les informations de configuration utilisées par IBM MQ MQI clients.

Pour plus d'informations, voir [IBM MQ le fichier de configuration du client MQI, mqclient.ini et l'emplacement du fichier de configuration du client.](#)

LDAP_BASEDN

LDAP_BASEDN est la variable d'environnement requise pour l'exécution d'un exemple de programme LDAP. Il indique le nom distinctif de base pour la recherche dans l'annuaire.

Hôte LDAP

LDAP_HOST est une variable d'environnement facultative pour l'exécution d'un exemple de programme LDAP. Il indique le nom de l'hôte sur lequel le serveur LDAP est exécuté ; il s'agit par défaut de l'hôte local s'il n'est pas spécifié.

VERSION LDAP

LDAP_VERSION est une variable d'environnement facultative pour l'exécution d'un exemple de programme LDAP. Il indique la version du protocole LDAP à utiliser et peut être 2 ou 3. La plupart des serveurs LDAP prennent désormais en charge la version 3 du protocole ; ils prennent tous en charge l'ancienne version 2. Cet exemple fonctionne également avec l'une ou l'autre version du protocole, et s'il n'est pas spécifié, il prend par défaut la version 2.

INTERVALLE_SUPPRESSE_CANAL_MQ_DISPONIBLE

La variable d'environnement **MQ_CHANNEL_SUPPRESS_INTERVAL** indique l'intervalle de temps, en secondes, pendant lequel les messages définis avec **MQ_CHANNEL_SUPPRESS_MSGS** ne doivent pas être écrits dans le journal des erreurs, ainsi que le nombre de fois où un message peut se produire pendant l'intervalle de temps spécifié avant d'être supprimé. La valeur par défaut est 60,5, ce qui signifie que toute autre occurrence d'un message donné est supprimée après les cinq premières occurrences de ce message dans un intervalle de 60 secondes. Pour plus d'informations, voir [Suppression des messages d'erreur de canal dans les journaux d'erreurs sur Multiplatforms.](#)

La variable d'environnement **MQ_CHANNEL_SUPPRESS_INTERVAL** est comparable à [SuppressInterval](#) dans le fichier `qm.ini`.

MQ_CHANNEL_SUPPRESS_MSGS

La variable d'environnement **MQ_CHANNEL_SUPPRESS_MSGS** supprime les messages d'erreur de canal dans le journal des erreurs. Vous pouvez spécifier une liste de messages supprimés.

MQ_CHANNEL_SUPPRESS_MSGS est utilisé conjointement avec **MQ_CHANNEL_SUPPRESS_INTERVAL**, qui indique le nombre de fois où chaque message apparaît avant d'être supprimé et la durée pendant laquelle

les messages sont supprimés. Pour plus d'informations, voir [Suppression des messages d'erreur de canal dans les journaux d'erreurs sur Multiplatforms](#).

La variable d'environnement **MQ_CHANNEL_SUPPRESS_MSGS** est comparable à `SuppressMessage` dans le fichier `qm.ini`, sauf que vous pouvez supprimer tout message de canal à l'aide de la variable d'environnement, alors qu'il existe une liste restrictive pour la méthode `qm.ini`.

TYPE MQ_CONNECT_



Sur Multiplatforms, vous pouvez utiliser la variable d'environnement **MQ_CONNECT_TYPE** en combinaison avec le type de liaison spécifié dans la zone Options de la structure MQCNO utilisée sur un appel MQCONN. **MQ_CONNECT_TYPE** n'a d'effet que pour les liaisons STANDARD. Pour les autres liaisons, **MQ_CONNECT_TYPE** est ignoré.

Pour plus d'informations, voir [Utilisation des options d'appel MQCONN avec MQ_CONNECT_TYPE](#).

MQ_CROSS_QUEUE_ORDER_ALL

Lorsque vous définissez la variable d'environnement **MQ_CROSS_QUEUE_ORDER_ALL** sur une valeur différente de zéro, l'ordre d'insertion de message est conservé dans une unité de travail. Cela signifie que si des messages d'une unité de travail (UoW) sont insérés dans plusieurs files d'attente (par exemple, Q1, puis Q2), lorsqu'un MQCMIT est émis, les messages sont distribués et mis à disposition dans le même ordre de file d'attente dans lequel ils ont été insérés.

Dans un environnement à plusieurs gestionnaires de files d'attente, **MQ_CROSS_QUEUE_ORDER_ALL** doit exister et avoir une valeur non vide côté émetteur et côté récepteur avant le démarrage de chaque gestionnaire de files d'attente.

PRÉFIXE ÉPHÉMÈRE MQ_

La variable d'environnement **MQ_EPHEMERAL_PREFIX** spécifie le chemin d'accès au répertoire éphémère du gestionnaire de files d'attente, dans lequel les données du gestionnaire de files d'attente éphémères sont conservées, pendant que le gestionnaire de files d'attente est en cours d'exécution.

Au lieu de modifier le préfixe éphémère en modifiant l'attribut **EphemeralPrefix** dans l'attribut **DefaultEphemeralPrefix** de la strophe AllQueueManagers du fichier `mqs.ini`, vous pouvez utiliser la variable d'environnement **MQ_EPHEMERAL_PREFIX** pour remplacer l'attribut **EphemeralPrefix** de la commande `crtmqm`. Pour plus d'informations, voir [Configurable ephemeral directory](#).

MQ_FILE_PATH



La variable d'environnement **MQ_FILE_PATH** est configurée lors de l'installation du package d'exécution sur la plateforme Windows. Cette variable d'environnement contient les mêmes données que la clé suivante dans le registre Windows :

```
HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation\InstallationName\FilePath
```

Pour plus d'informations, voir [setmqenv \(set IBM MQ environment\)](#) et [crtmqenv \(create IBM MQ environment\)](#).

MQ_JAVA_DATA_PATH

La variable d'environnement **MQ_JAVA_DATA_PATH** spécifie le répertoire de la sortie de journal et de trace pour IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging et IBM MQ classes for Java. Il est utilisé par les scripts fournis avec IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging et IBM MQ classes for Java.

Pour plus d'informations, voir [Définition des variables d'environnement pour les classes IBM MQ pour JMS/Jakarta Messaging](#) et les variables d'environnement [relatives aux classes IBM MQ pour Java](#).

MQ_JAVA_INSTALL_PATH

La variable d'environnement **MQ_JAVA_INSTALL_PATH** indique le répertoire dans lequel sont installés les IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging , comme indiqué dans la rubrique [What is installed for IBM MQ classes for JMS](#), et les IBM MQ classes for Java , comme indiqué dans les répertoires d'installation [IBM MQ classes for Java](#) .

Pour plus d'informations, voir [Définition des variables d'environnement pour les classes IBM MQ pour JMS/Jakarta Messaging](#) et les variables d'environnement [relatives aux classes IBM MQ pour Java](#).

MQ_JAVA_LIB_PATH

La variable d'environnement **MQ_JAVA_LIB_PATH** indique le répertoire dans lequel sont stockées les bibliothèques IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging, ainsi que les bibliothèques IBM MQ classes for Java . Certains scripts, par exemple IVTRun, qui sont fournis avec IBM MQ classes for JMS et IBM MQ classes for Jakarta Messaging ou IBM MQ classes for Java utilisent cette variable d'environnement.

Pour plus d'informations, voir [Définition des variables d'environnement pour les classes IBM MQ pour JMS/Jakarta Messaging](#) et les variables d'environnement [relatives aux classes IBM MQ pour Java](#).

CHEMIN_DATA_OVERRIDE_MQ

Vous pouvez utiliser la variable d'environnement **MQ_OVERRIDE_DATA_PATH** pour modifier le répertoire par défaut du chemin de données IBM MQ .

MQ_SET_NODELAYACK

AIX

La variable d'environnement **MQ_SET_NODELAYACK** désactive l'accusé de réception retardé TCP sur AIX.

Lorsque vous définissez cette variable d'environnement, le paramètre désactive l'accusé de réception retardé TCP en appelant l'appel `setsockopt` du système d'exploitation avec l'option `TCP_NODELAYACK` . Seul AIX prend en charge cette fonction, de sorte que la variable d'environnement **MQ_SET_NODELAYACK** n'a d'effet que sur AIX.

NOM_UTILISATEUR_MQ

Linux

Vous pouvez utiliser la variable d'environnement **MQ_USER_NAME** pour permettre à une installation non enregistrée sur Linux de choisir le nom d'un utilisateur sans nom. Cela est nécessaire, par exemple, pour utiliser des hiérarchies de publication / abonnement dans OpenShift.

La valeur de **MQ_USER_NAME** ne doit pas correspondre à un utilisateur préexistant sur le système et doit être inférieure ou égale à 12 octets.

MQAPI_XX_ENCODE_CASE_ONE fichier_journal_trace

L'exemple de programme d'exit API génère une trace MQI dans un fichier spécifié par l'utilisateur avec un préfixe défini dans la variable d'environnement **MQAPI_TRACE_LOGFILE** .

Pour plus d'informations, voir [L'exemple de programme d'exit d'API](#).

MQAPPLNAME

ALW

Si le nom de l'application n'a pas encore été choisi, vous pouvez utiliser la variable d'environnement **MQAPPLNAME** comme nom à utiliser pour identifier la connexion à un gestionnaire de files d'attente. Seuls les 28 premiers caractères sont utilisés et ne doivent pas être des blancs ou des valeurs nulles.

Pour plus d'informations, voir [Utilisation du nom d'application dans les langages de programmation pris en charge](#).

MQCCSID

La variable d'environnement **MQCCSID** indique le numéro de jeu de caractères codés à utiliser et remplace la valeur CCSID avec laquelle le serveur a été configuré. **MQCCSID** peut être utilisé pour remplacer le CCSID natif d'une application et spécifier le numéro de jeu de caractères codés à utiliser, par exemple si le CCSID natif est un CCSID non pris en charge ou n'est pas le CCSID requis.

Pour définir **MQCCSID**, utilisez l'une des commandes suivantes:

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

```
export MQCCSID=number
```

- **Windows** Sous Windows :

```
SET MQCCSID=number
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQCCSID) VALUE(number)
```

Pour plus d'informations, voir [Choix du CCSID client ou serveur](#).

URL MQCCDTURL

La variable d'environnement **MQCCDTURL** offre la capacité équivalente à la définition d'une combinaison des variables d'environnement **MQCHLLIB** et **MQCHLTAB**. Il vous permet de fournir un fichier, une URL ftp ou une URL http sous la forme d'une valeur unique à partir de laquelle une table de définition de canal du client peut être obtenue pour les programmes natifs se connectant en tant que clients, c'est-à-dire des applications C, COBOL ou C++.

Remarque : L'utilisation de variables d'environnement pour fournir l'URL n'a aucun effet sur les applications Java, JMS ou .NET gérées.

IBM MQ prend en charge l'extraction d'une table de définition de canal du client à partir d'un fichier, d'une adresse FTP ou d'une URL HTTP. Toutefois, **MQCCDTURL** n'accepte qu'une valeur d'URL. Il n'accepte pas le format de répertoire du système de fichiers local existant.

Pour utiliser **MQCCDTURL** à la place de **MQCHLLIB** et **MQCHLTAB** avec un fichier local, vous pouvez utiliser un protocole 'file://'. Par conséquent, comme illustré dans cet exemple pour AIX et Linux:

```
export MQCCDTURL=file:///var/mqm/qmgrs/QMGR/@ipcc/MYCHL.TAB
```

équivalent à :

```
export MQCHLLIB=/var/mqm/qmgrs/QMGR/@ipcc
export MQCHLTAB=MYCHL.TAB
```

Vous pouvez également spécifier un fichier JSON comme illustré dans cet exemple pour Windows:

```
set MQCCDTURL=file:/c:/mq-channels/CCDT-QMGR1.json
```

équivalent à :

```
set MQCHLLIB=C:\mq-channels
set MQCHLTAB=CCDT-QMGR1.json
```

Pour plus d'informations, voir [Accès par URL à la table de définition de canal du client](#).

MQCERTLABL

La variable d'environnement **MQCERTLABL** définit le libellé de certificat d'une définition de canal à utiliser par IBM MQ pour localiser un certificat personnel envoyé lors de l'établissement d'une liaison TLS.

Pour plus d'informations, voir [Labels de certificat numérique, compréhension des exigences](#).

MQCERTVPOL

La variable d'environnement **MQCERTVPOL** indique le type de règle de validation de certificat à utiliser. Cette variable d'environnement remplace l'attribut **CertificateValPolicy** dans la strophe SSL du fichier de configuration du client.

MQCERTVPOL peut être défini sur l'une des deux valeurs suivantes:

ANY

Utilisez toute règle de validation de certificat prise en charge par la bibliothèque de sockets sécurisés sous-jacente. Il s'agit du paramètre par défaut.

RFC5280

Utilisez uniquement la validation de certificat conforme à la norme RFC 5280.

Pour définir **MQCERTVPOL**, utilisez l'une des commandes suivantes:

-   Pour les systèmes AIX and Linux :

```
export MQCERTVPOL= value
```

-  Pour les systèmes Windows :

```
SET MQCERTVPOL= value
```

-  Pour les systèmes IBM i :

```
ADDENVVAR ENVVAR(MQCERTVPOL) VALUE(value)
```

Pour plus d'informations, voir [Certificate validation policies in IBM MQ](#) and [Configuring certificate validation policies in IBM MQ](#).

MQCHLLIB

La variable d'environnement **MQCHLLIB** indique le chemin de répertoire du fichier qui contient la table de définition de canal du client (CCDT). Le fichier est créé sur le serveur, mais il peut être copié sur le poste de travail IBM MQ MQI client .

Pour définir **MQCHLLIB**, utilisez l'une des commandes suivantes:

-  Sous Windows :

```
SET MQCHLLIB=pathname
```

Exemple :

```
SET MQCHLLIB=C:\wmqtest
```

- **Linux** **AIX** Pour les systèmes AIX and Linux :

```
export MQCHLLIB=pathname
```

- **IBM i** Pour IBM i :

```
ADDENVVAR ENVVAR(MQCHLLIB) VALUE(pathname)
```

Si **MQCHLLIB** n'est pas défini, le chemin du client est par défaut:

- **Linux** **AIX** Sous AIX and Linux : `/var/mqm/`
- **Windows** Sous Windows : `MQ_INSTALLATION_PATH`
- **IBM i** Sous IBM i : `/QIBM/UserData/mqm/`

Pour les commandes **crtmqm** et **strmqm**, le chemin d'accès par défaut est l'un des deux ensembles de chemins d'accès. Si *datapath* est défini, le chemin d'accès par défaut est l'un des premiers ensembles. Si *datapath* n'est pas défini, le chemin d'accès par défaut est l'un des deuxièmes ensembles.

- **Linux** **AIX** Sous AIX and Linux : `datapath/@ipcc`
- **Windows** Sous Windows : `datapath\@ipcc`
- **IBM i** Sous IBM i : `datapath/&ipcc`

Ou :

- **Linux** **AIX** Sous AIX and Linux : `/prefix/qmgrs/qmgrname/@ipcc`
- **Windows** Sous Windows : `MQ_INSTALLATION_PATH\data\qmgrs\qmgrname\@ipcc`
- **IBM i** Sous IBM i : `/prefix/qmgrs/qmgrname/&ipcc`

où :

- `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.
- S'il est présent, *datapath* est la valeur de DataPath définie dans la section du gestionnaire de files d'attente.
- *prefix* est la valeur du préfixe défini dans la section du gestionnaire de files d'attente. Le préfixe est généralement l'une des valeurs suivantes:
 - **Linux** **AIX** `/var/mqm` sur les systèmes AIX and Linux .
 - **IBM i** `/QIBM/UserData/mqm/` sur IBM i.
- *qmgrname* est la valeur de l'attribut Directory défini dans la section du gestionnaire de files d'attente. La valeur peut être différente du nom réel du gestionnaire de files d'attente. Il se peut que la valeur ait été modifiée pour remplacer des caractères spéciaux.
- L'emplacement où la section du gestionnaire de files d'attente est définie dépend de la plateforme:
 - **IBM i** **Linux** **AIX** Dans le fichier `mq.s.ini` sous IBM i, AIX and Linux.
 - **Windows** Dans le registre sur Windows.

Remarques :

1. **z/OS** Si vous utilisez IBM MQ for z/OS comme serveur, le fichier doit être conservé sur le poste de travail client IBM MQ .

2. Si ce paramètre est défini, MQCHLLIB remplace le chemin utilisé pour localiser la table de définition de canal du client.
3. MQCHLLIB peut contenir une URL qui fonctionne en combinaison avec la variable d'environnement MQCHLTAB (voir «URL d'accès à la table de définition de canal du client», à la page 56).
4. Les variables d'environnement, telles que **MQCHLLIB**, peuvent être sectorisées à un processus ou à un travail, ou à l'échelle du système, d'une manière spécifique à la plateforme.
5. Si vous définissez **MQCHLLIB** à l'échelle du système sur un serveur, il définit le même chemin d'accès au fichier CCDT pour tous les gestionnaires de files d'attente sur le serveur. Si vous ne définissez pas la variable d'environnement **MQCHLLIB**, le chemin d'accès est différent pour chaque gestionnaire de files d'attente. Les gestionnaires de files d'attente lisent la valeur de **MQCHLLIB**, si elle est définie, sur la commande **crtmqm** ou **strmqm**.
6. Si vous créez plusieurs gestionnaires de files d'attente sur un serveur, la distinction est importante, pour la raison suivante. Si vous définissez **MQCHLLIB** à l'échelle du système, chaque gestionnaire de files d'attente met à jour le même fichier CCDT. Le fichier contient les définitions de connexion client de tous les gestionnaires de files d'attente sur le serveur. Si la même définition existe sur plusieurs gestionnaires de files d'attente, SYSTEM.DEF.CLNTCONN par exemple, le fichier contient la définition la plus récente. Lorsque vous créez un gestionnaire de files d'attente, si **MQCHLLIB** est défini, SYSTEM.DEF.CLNTCONN est mis à jour dans la table de définition de canal du client. La mise à jour remplace le fichier SYSTEM.DEF.CLNTCONN créé par un autre gestionnaire de files d'attente. Si vous avez modifié la définition précédente, vos modifications sont perdues. Pour cette raison, vous devez envisager de trouver des alternatives à la définition de **MQCHLLIB** en tant que variable d'environnement à l'échelle du système sur le serveur.
7. L'option NOREPLACE de MQSC et PCF sur une définition de connexion client ne vérifie pas le contenu du fichier CCDT. Une définition de canal de connexion client du même nom que celle qui a été créée précédemment, mais pas par ce gestionnaire de files d'attente, est remplacée, quelle que soit l'option NOREPLACE. Si la définition a été précédemment créée par le même gestionnaire de files d'attente, elle n'est pas remplacée.
8. La commande **rcrmqobj -t clchltab** supprime et recrée le fichier CCDT. Le fichier est recréé avec uniquement les définitions de connexion client créées sur le gestionnaire de files d'attente sur lequel la commande est exécutée.
9. Les autres commandes qui mettent à jour la table de définition de canal du client modifient uniquement les canaux de connexion client ayant le même nom de canal. Les autres canaux de connexion client du fichier ne sont pas modifiés.
10. Le chemin d'accès à **MQCHLLIB** n'a pas besoin de guillemets.

Pour plus d'informations, voir [Emplacements de la table de définition de canal du client](#), [Accès URL à la table de définition de canal du client](#) et [Connexion des applications client aux gestionnaires de files d'attente à l'aide de variables d'environnement](#).

MQCHLTAB

La variable d'environnement **MQCHLTAB** indique le nom du fichier qui contient la table de définition de canal du client (CCDT). Le nom de fichier par défaut est AMQCLCHL.TAB.

Pour définir **MQCHLTAB**, utilisez l'une des commandes suivantes:

-   Sous AIX and Linux :

```
export MQCHLTAB=filename
```

-  Sous Windows :

```
SET MQCHLTAB=filename
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQCHLTAB) VALUE(filename)
```

Exemple :

```
SET MQCHLTAB=ccdf1.tab
```

De la même manière que pour le client, la variable d'environnement **MQCHLTAB** sur le serveur spécifie le nom de la table de définition de canal du client.

Pour plus d'informations, voir [Emplacements de la table de définition de canal du client](#), [Accès URL à la table de définition de canal du client](#) et [Connexion des applications client aux gestionnaires de files d'attente à l'aide de variables d'environnement](#).

MQCLNTCF

La variable d'environnement **MQCLNTCF** indique l'emplacement du fichier de configuration IBM MQ MQI client . Ce fichier contient les informations de configuration utilisées par IBM MQ MQI clients.

Vous pouvez utiliser la variable d'environnement **MQCLNTCF** pour modifier le chemin du fichier `mqclient.ini`.

Le format de cette variable d'environnement est une URL complète. Cela signifie que le nom de fichier n'est pas nécessairement `mqclient.ini`, ce qui facilite le placement du fichier sur un système de fichiers connecté au réseau. Pour plus d'informations, voir [IBM MQ le fichier de configuration du client MQI](#), `mqclient.ini` et [l'emplacement du fichier de configuration du client](#).

MQDOTNET_TRACE_ON

La variable d'environnement **MQDOTNET_TRACE_ON** est utilisée pour activer la trace pour les clients redistribuables IBM MQ .NET . Les valeurs inférieures ou égales à 0 n'activent pas la trace, 1 active la trace par défaut et les valeurs supérieures à 1 activent la fonction de trace détaillée.

Pour plus d'informations, voir [Traçage des applications IBM MQ .NET](#) et [Traçage des applications IBM MQ .NET à l'aide de variables d'environnement](#).

MQIPADDRV

La variable d'environnement **MQIPADDRV** indique le protocole IP à utiliser pour une connexion de canal. Il peut avoir les valeurs de chaîne "MQIPADDR_IPV4" ou "MQIPADDR_IPV6". Ces valeurs ont la même signification que IPv4 et IPv6 dans [ALTER QMGR IPADDRV](#) et l'attribut **IPAddressVersion** de la strophe TCP du fichier de configuration du client. Si la variable d'environnement n'est pas définie, "MQIPADDR_IPV4" est supposé.

Pour définir **MQIPADDRV**, utilisez l'une des commandes suivantes:

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

```
export MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6" />
```

- **Windows** Sous Windows :

```
SET MQIPADDRV=MQIPADDR_IPV4|MQIPADDR_IPV6
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQIPADDRV) VALUE(MQIPADDR_IPV4|MQIPADDR_IPV6)
```

Mot de passe MQKEYRPWD

Lorsque vous définissez la variable d'environnement *MQKEYRPWD*, elle indique le mot de passe du référentiel de clés qui contient le certificat numérique appartenant à l'utilisateur. Si *MQKEYRPWD* est utilisé, vous devez chiffrer le mot de passe avant de définir la valeur de la variable d'environnement.

Pour définir *MQKEYRPWD*, utilisez l'une des commandes suivantes:

- **Linux** **AIX** Sur les systèmes AIX and Linux :

```
export MQKEYRPWD=passphrase
```

- **Windows** Sur les systèmes Windows :

```
SET MQKEYRPWD=passphrase
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQKEYRPWD) VALUE(passphrase)
```

Il n'existe pas de valeur par défaut pour cette variable d'environnement.

Pour plus d'informations, voir

- **ALW** Définition du mot de passe du référentiel de clés pour un IBM MQ MQI client sur AIX, Linux, and Windows et [Chiffrement du mot de passe du référentiel de clés](#)
- **IBM i** Indiquer le mot de passe du référentiel de clés pour un IBM MQ MQI client sur IBM i et [Chiffrer le mot de passe du référentiel de clés](#).

mqlicense

Linux

Sur les systèmes Linux, vous pouvez utiliser la variable d'environnement **MQLICENSE** pour accepter ou afficher une licence IBM MQ après avoir installé le produit.

Pour plus d'informations sur la raison pour laquelle vous souhaitez ou devez effectuer cette opération, voir [Acceptation de la licence sur IBM MQ for Linux](#)

La variable d'environnement **MQLICENSE** peut être définie sur l'une des deux valeurs suivantes:

accepter

Acceptez la post-installation de la licence.

afficher

Affichez la licence, si elle a été acceptée.

Pour accepter la post-installation de la licence, utilisez la commande suivante:

```
export MQLICENSE=accept
```

Pour afficher la licence, utilisez la commande suivante:

```
export MQLICENSE=view
```


Remarque : Vous pouvez également utiliser les commandes suivantes pour accepter et afficher la licence:

- [mqlicense](#) (acceptation de la licence après l'installation)
- [dspmqlic](#) (affichage de la licence IBM MQ)

MQMAXERRORLOGSIZE

Multi

La variable d'environnement **MQMAXERRORLOGSIZE** indique la taille du journal des erreurs du gestionnaire de files d'attente qui est copié dans la sauvegarde.

Pour plus d'informations, voir [Utilisation des journaux d'erreurs](#).

NOM MQE

Windows

La variable d'environnement **MQNAME** indique le nom NetBIOS local que les processus IBM MQ peuvent utiliser. Une connexion NetBIOS s'applique uniquement à un client et un serveur exécutant Windows.

Pour définir **MQNAME**, utilisez la commande suivante:

```
SET MQNAME=Your_env_Name
```

Exemple :

```
SET MQNAME=CLIENT1
```

Certaines implémentations NetBIOS requièrent un nom unique, défini par **MQNAME**, pour chaque application si vous exécutez simultanément plusieurs applications IBM MQ sur le IBM MQ MQI client.

Pour plus d'informations, voir [Définition du nom IBM MQ local NetBIOS](#).

MQNOREMPOOL

Lorsque vous définissez la variable d'environnement **MQNOREMPOOL**, elle désactive le regroupement de canaux et entraîne l'exécution des canaux en tant qu'unités d'exécution du programme d'écoute.

Pour plus d'informations, voir [MCATYPE](#) (type d'agent de canal de message).

FICHER JOURNAL MQPSE_TRACE_

Vous utilisez la variable d'environnement **MQPSE_TRACE_LOGFILE** lorsque vous exécutez l'exemple de programme d'exit de publication AMQSPSE0, qui est un exemple de programme C d'un exit pour intercepter une publication avant qu'elle ne soit distribuée à un abonné. Dans le processus d'application à tracer, cette variable d'environnement décrit l'emplacement dans lequel les fichiers de trace doivent être écrits.

Pour plus d'informations, voir [L'exemple de programme d'exit de publication](#).

MQS_AMSCRED_KEYFILE

Vous pouvez utiliser la variable d'environnement **MQS_AMSCRED_KEYFILE** pour remplacer ou fournir le fichier de clés initial à utiliser lors de l'exécution des applications IBM MQ Advanced Message Security (AMS) ou lorsque vous protégez un fichier de configuration de magasin de clés à l'aide de la commande **runamscred**.

Pour plus d'informations, voir [Utilisation de magasins de clés et de certificats avec AMS](#) et [Protection des mots de passe dans les fichiers de configuration du composant IBM MQ](#).

MQS_DISABLE_ALL_INTERCEPT

Vous pouvez utiliser la variable d'environnement **MQS_DISABLE_ALL_INTERCEPT** pour désactiver IBM MQ Advanced Message Security (AMS) si une erreur 2085 (MQRC_UNKNOWN_OBJECT_NAME) est signalée lorsque vous tentez de vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit et que vous utilisez IBM MQ avec des clients C natifs.

Remarque : Vous pouvez utiliser la variable d'environnement **MQS_DISABLE_ALL_INTERCEPT** pour les clients C uniquement. Pour les clients Java , vous devez utiliser la variable d'environnement **AMQ_DISABLE_CLIENT_AMS** à la place.

Pour plus d'informations, voir [Désactivation d'Advanced Message Security sur le client](#).

MQS_IPC_HOTE

Etant donné que les objets du système de fichiers IPC doivent être distingués par le système, un sous-répertoire pour chaque système sur lequel s'exécute le gestionnaire de files d'attente est ajouté au chemin de répertoire. Si la valeur générée du nom d'hôte crée un problème, vous pouvez définir le nom d'hôte à l'aide de la variable d'environnement **MQS_IPC_HOST** .

Pour plus d'informations, voir [Partage de fichiers IBM MQ sur Multiplatforms](#).

MQS_KEYSTORE_CONF

La variable d'environnement **MQS_KEYSTORE_CONF** indique l'emplacement du fichier de configuration du magasin de clés pour IBM MQ Advanced Message Security (AMS), si le fichier ne se trouve pas à l'emplacement par défaut de *home_directory/.mq/keystore.conf*.

Pour plus d'informations, voir [Utilisation de magasins de clés et de certificats avec AMS](#).

Si vous rencontrez des problèmes sous Managed File Transfer, voir [Traitement des incidents lorsque MFT ne lit pas les propriétés de magasin de clés pour AMS](#).

FICHER_CLE_MQS_MQI

Lorsque vous définissez la variable d'environnement **MQS_MQI_KEYFILE** , elle indique l'emplacement d'un fichier de clés initial qui contient la clé initiale à utiliser pour les opérations de protection par mot de passe. Si le fichier de clés initial n'est pas spécifié, la clé initiale par défaut est utilisée par le système de protection par mot de passe IBM MQ .


Pour définir **MQS_MQI_KEYFILE**, utilisez l'une des commandes suivantes:

-   Sur les systèmes AIX and Linux :

```
export MQS_MQI_KEYFILE=key file location
```

-  Sur les systèmes Windows :

```
SET MQS_MQI_KEYFILE=key file location
```

-  Sous IBM i :

```
ADDENVVAR ENVVAR(MQS_MQI_KEYFILE) VALUE(key file location)
```

Pour plus d'informations, voir [Fourniture d'une clé initiale pour un IBM MQ MQI client sur AIX, Linux, and Windows](#) et [Fourniture d'une clé initiale pour un IBM MQ MQI client sur IBM i](#).

MQS_SSLCRYP_KEYFILE

La variable d'environnement **MQS_SSLCRYP_KEYFILE** est une autre façon de spécifier le chemin d'accès complet et le nom du fichier contenant la clé initiale utilisée pour chiffrer le mot de passe dans la chaîne de configuration du matériel de cryptographie PKCS #11 , au lieu de la spécifier avec l'attribut

SSLCryptoHardwareKeyFile dans la strophe SSL du `qm.ini`. **MQS_SSLCRYP_KEYFILE** ayant une priorité supérieure à celle du fichier `qm.ini`, sa valeur est prioritaire sur toute autre valeur. Pour plus d'informations, voir [Clients IBM MQ utilisant du matériel de cryptographie](#).

OPTIONS-TRACE_MQ

AIX

Pour la fonction de trace de composant sélectif sous AIX, utilisez la variable d'environnement **MQS_TRACE_OPTIONS** pour activer individuellement les fonctions de trace de paramètre et de détail élevé.

Remarque : Ne définissez la variable d'environnement **MQS_TRACE_OPTIONS** que si vous avez été invité à le faire par le support IBM.

Pour plus d'informations, voir [Fonction de trace sur AIX and Linux](#).

MQSERVER

La variable d'environnement **MQSERVER** est utilisée pour définir un canal minimal. **MQSERVER** indique l'emplacement du serveur IBM MQ et la méthode de communication à utiliser.

Remarque : Vous ne pouvez pas utiliser **MQSERVER** pour définir un canal TLS ou un canal avec des exits de canal. Pour plus d'informations sur la définition d'un canal TLS, voir [Protection des canaux avec TLS](#).

Les exemples suivants montrent comment définir **MQSERVER**:

- Linux AIX Sous AIX and Linux :

```
export MQSERVER='SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)'
```

- Windows Sous Windows :

```
SET MQSERVER=SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)
```

- IBM i Sous IBM i :

```
ADDENVVAR ENVVAR(MQSERVER) VALUE('SYSTEM.DEF.SVRCONN/TCP/AMACHINE.ACOMPANY.COM(1414)')
```

Remarque :

- Le nom de canal ne peut pas contenir de barre oblique (/) car ce caractère est utilisé pour séparer le nom de canal, le type de transport et le nom de connexion. Lorsque la variable d'environnement **MQSERVER** est utilisée pour définir un canal client, une longueur maximale de message (MAXMSGL) de 100 Mo est utilisée. Par conséquent, la taille de message maximale en vigueur pour le canal est la valeur spécifiée dans le canal SVRCONN sur le serveur.
- Le type de transport peut être LU62, TCP, NETBIOS, SPX, selon votre plateforme client IBM MQ.
- Le nom de connexion doit être un nom de réseau qualifié complet. Par exemple, AMACHINE.ACOMPANY.COM(1414).
- Le nom de connexion peut être une liste de noms de connexion séparés par des virgules. Les noms de connexion de la liste sont utilisés de la même manière que plusieurs connexions dans une table de connexion client. La liste de noms de connexion peut être utilisée comme alternative aux groupes de gestionnaires de files d'attente pour spécifier plusieurs connexions que le client doit essayer. Si vous configurez un gestionnaire de files d'attente multi-instance, vous pouvez utiliser une liste de noms de connexion pour spécifier des instances de gestionnaire de files d'attente différentes.

Si vous utilisez la variable d'environnement **MQSERVER** pour définir le canal entre votre machine IBM MQ MQI client et une machine serveur, il s'agit du seul canal disponible pour votre application et aucune référence n'est faite à la table de définition de canal du client (CCDT).

Pour plus d'informations, voir [Création d'un canal de connexion client sur le client IBM MQ MQI à l'aide de MQSERVER](#).

MQSNOAUT



Avertissement : Cette fonctionnalité n'est pas recommandée.

Lorsque vous définissez la variable d'environnement **MQSNOAUT** sur une valeur quelconque, elle désactive le gestionnaire des droits d'accès aux objets (OAM) et empêche toute vérification de la sécurité. Cela peut convenir à un environnement de test. Cela inclut à la fois la fonctionnalité d'autorisation et d'authentification de connexion. TLS, les enregistrements d'authentification de canal et les exits de sécurité ne sont pas affectés.

La variable d'environnement **MQSNOAUT** prend effet uniquement lorsqu'un gestionnaire de files d'attente est créé.



Avertissement : Pour activer la méthode d'accès aux objets (OAM), vous devez supprimer le gestionnaire de files d'attente, supprimer la variable d'environnement, puis recréer le gestionnaire de files d'attente sans spécifier **MQSNOAUT**.

Pour plus d'informations, voir [Prévention des contrôles d'accès de sécurité sur les systèmes AIX, Linux et Windows](#).

MQSPREFIX

Au lieu de changer le préfixe par défaut, vous pouvez utiliser la variable d'environnement **MQSPREFIX** pour remplacer le préfixe **DefaultPrefix** de la commande **crtmqm**.

Pour plus d'informations, voir [IBM MQ file names](#) et [AllQueueManagers strophe du fichier mqs.ini](#).

MQSSLCRYP




La variable d'environnement **MQSSLCRYP** contient une chaîne de paramètres que vous pouvez utiliser pour configurer le matériel cryptographique présent sur le système.

Les valeurs admises sont les mêmes que pour la zone [SSLCryptoHardware](#) dans la strophe SSL du fichier de configuration du client.

Pour définir **MQSSLCRYP**, utilisez l'une des commandes suivantes:

-   Sur les systèmes AIX and Linux :

```
export MQSSLCRYP=string
```

-  Sur les systèmes Windows :

```
SET MQSSLCRYP=string
```

Pour plus d'informations, voir [Configuring for cryptographic hardware on AIX, Linux, and Windows and IBM MQ clients using cryptographic hardware](#) dans la rubrique [Protection des mots de passe dans les fichiers de configuration de composant IBM MQ](#).

MQSSLFIPS

La variable d'environnement **MQSSLFIPS** indique si seuls les algorithmes certifiés FIPS doivent être utilisés si la cryptographie est effectuée dans IBM MQ. Vous pouvez définir cette variable d'environnement sur YES ou NO. La valeur par défaut est NO. Ces valeurs sont identiques à celles du paramètre **SSLFIPS** de la commande [ALTER QMGR](#).

Pour définir **MQSSLFIPS**, utilisez l'une des commandes suivantes:

- **Linux** **AIX** Sur les systèmes AIX and Linux :

```
export MQSSLFIPS=YES|NO
```

- **Windows** Sur les systèmes Windows :

```
SET MQSSLFIPS=YES|NO
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQSSLFIPS) VALUE(YES|NO)
```

L'utilisation d'algorithmes certifiés FIPS est affectée par l'utilisation de matériel de cryptographie. Pour plus d'informations, voir [Spécification du fait que seuls les CipherSpecs certifiés FIPS sont utilisés lors de l'exécution sur le client MQI](#).

MQSSLKEYR

La variable d'environnement **MQSSLKEYR** indique l'emplacement du référentiel de clés qui contient le certificat numérique appartenant à l'utilisateur.

Indiquez le chemin d'accès complet et le nom de fichier du référentiel de clés. Si le suffixe de fichier n'est pas spécifié, il est supposé être `.kdb`.

Pour définir **MQSSLKEYR**, utilisez l'une des commandes suivantes:

- **Linux** **AIX** Sur les systèmes AIX and Linux :

```
export MQSSLKEYR=pathname
```

- **Windows** Sur les systèmes Windows :

```
SET MQSSLKEYR=pathname
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQSSLKEYR) VALUE(pathname)
```

Il n'existe pas de valeur par défaut pour cette variable d'environnement.

Pour plus d'informations, voir le paramètre **SSLKEYR** de la commande [ALTER QMGR](#).

MQSSLPROXY

La variable d'environnement **MQSSLPROXY** indique le nom d'hôte et le numéro de port du serveur proxy HTTP à utiliser par GSKit pour les vérifications OCSP.

Pour définir **MQSSLPROXY**, utilisez l'une des commandes suivantes:

- **Linux** **AIX** Sur les systèmes AIX and Linux :

```
export MQSSLPROXY="string"
```

- **Windows** Sur les systèmes Windows :

```
SET MQSSLPROXY= string
```

La chaîne que vous spécifiez avec **MQSSLPROXY** peut être le nom d'hôte ou l'adresse réseau du serveur proxy HTTP qui doit être utilisé par GSKit pour les vérifications OCSP. Cette adresse peut être suivie d'un numéro de port facultatif, entre parenthèses. Si vous n'indiquez pas le numéro de port, le port HTTP par défaut, 80, est utilisé.

- **Linux** **AIX** Par exemple, sur les systèmes AIX and Linux , vous pouvez utiliser l'une des commandes suivantes:

- ```
export MQSSLPROXY="proxy.example.com(80) "
```

- ```
export MQSSLPROXY="127.0.0.1"
```

Pour plus d'informations, voir [Utilisation du protocole OCSP \(Online Certificate Status Protocol\)](#).

MQSSLRESET

La variable d'environnement **MQSSLRESET** indique le nombre d'octets non chiffrés envoyés et reçus sur un canal TLS avant la renégociation de la clé secrète TLS. Il peut être défini sur un entier compris entre 0 et 999 999 999. La valeur par défaut est 0, ce qui indique que les clés secrètes ne sont jamais renégociées. Si vous spécifiez un nombre de réinitialisations de clé confidentielle TLS compris entre 1 octet et 32 Ko, les canaux TLS utilisent un nombre de réinitialisations de clé confidentielle de 32 Ko. Ce nombre de réinitialisations de clé secrète permet d'éviter un nombre excessif de réinitialisations de clé qui se produiraient pour les petites valeurs de réinitialisation de clé secrète TLS.

Pour définir **MQSSLRESET**, utilisez l'une des commandes suivantes:

- **Linux** **AIX** Sur les systèmes AIX and Linux :

```
export MQSSLRESET=integer
```

- **Windows** Sur les systèmes Windows :

```
SET MQSSLRESET=integer
```

- **IBM i** Sous IBM i :

```
ADDENVVAR ENVVAR(MQSSLRESET) VALUE(integer)
```

Pour plus d'informations, voir [Réinitialisation des clés secrètes SSL et TLS](#).

MQSUITEB

ALW

Vous pouvez configurer IBM MQ pour qu'il fonctionne conformément à la norme NSA Suite B sur les plateformes AIX, Linux, and Windows .

La variable d'environnement **MQSUITEB** indique si la cryptographie compatible Suite B doit être utilisée. Si la cryptographie Suite B doit être utilisée, vous pouvez spécifier la puissance de la cryptographie en définissant **MQSUITEB** sur l'une des valeurs suivantes:

- Aucun

- 128_BIT, 192_BIT
- 128_BIT
- 192_BIT

Vous pouvez spécifier plusieurs valeurs à l'aide d'une liste séparée par des virgules. L'utilisation de la valeur NONE avec une autre valeur n'est pas valide.

Pour plus d'informations, voir [Configuration de IBM MQ pour Suite B](#).

MQTCPTIMEOUT

La variable d'environnement **MQTCPTIMEOUT** indique la durée pendant laquelle IBM MQ attend un appel de connexion TCP.

MSG ODQ

Si vous utilisez un gestionnaire de files d'attente de rebut différent de **runmqdlq**, la source de l'exemple, **amqsd1q**, est disponible pour vous permettre de l'utiliser comme base. L'exemple est similaire au gestionnaire de rebut fourni dans le produit, mais les rapports de trace et d'erreur sont différents. Utilisez la variable d'environnement **ODQ_MSG** pour définir le nom du fichier contenant les messages d'erreur et d'information. Le fichier fourni est appelé **amqsd1q.msg**.

Pour plus d'informations, voir [Exemple de gestionnaire de files d'attente de rebut](#).

TRACE ODQ_

Si vous utilisez un gestionnaire de files d'attente de rebut différent de **runmqdlq**, la source de l'exemple, **amqsd1q**, est disponible pour vous permettre de l'utiliser comme base. L'exemple est similaire au gestionnaire de rebut fourni dans le produit, mais les rapports de trace et d'erreur sont différents. Pour activer la fonction de trace, définissez la variable d'environnement **ODQ_TRACE** sur YES ou yes.

Pour plus d'informations, voir [Exemple de gestionnaire de files d'attente de rebut](#).

WCF_TRACE_ON

Deux méthodes de trace différentes sont disponibles pour le canal personnalisé WCF. Ces deux méthodes de trace sont activées indépendamment ou ensemble. Chaque méthode produit son propre fichier de trace, de sorte que lorsque les deux méthodes de trace ont été activées, deux fichiers de sortie de trace sont générés. Il existe quatre combinaisons pour activer et désactiver les deux méthodes de trace différentes. Outre ces combinaisons permettant d'activer la trace WCF, la trace XMS .NET peut être activée à l'aide de la variable d'environnement **WCF_TRACE_ON**.

Pour plus d'informations, voir [Traçage du canal personnalisé WCF pour IBM MQ](#).

REP_BASE_WMQSO

La variable d'environnement **WMQSOAP_HOME** est utilisée lors de l'exécution d'étapes de configuration supplémentaires après l'installation et la configuration correctes de l'environnement d'hébergement de service .NET SOAP sur JMS dans IBM MQ. Il est accessible à partir d'un gestionnaire de files d'attente local.

Pour plus d'informations, voir [WCF client to a .NET service hosted by IBM MQ sample](#) et [WCF client to an Axis Java service hosted by IBM MQ sample](#).

XMS_TRACE_ON, XMS_TRACE_FILE_PATH, XMS_TRACE_FORMAT et XMS_TRACE_SPECIFICATION

Si vous utilisez IBM MQ classes for XMS .NET Framework, vous pouvez configurer la trace à partir d'un fichier de configuration d'application ainsi qu'à partir des variables d'environnement XMS. Si vous utilisez des IBM MQ classes for XMS .NET (bibliothèques.NET Standard et .NET 6), vous devez configurer la trace

à partir des variables d'environnement XMS . En général, la trace est utilisée sur les conseils du support IBM.

Pour activer et configurer la trace pour une application XMS .NET , définissez les variables d'environnement suivantes avant d'exécuter l'application:

XMS_TRACE_ON

Si la variable d'environnement **XMS_TRACE_ON** est définie, toutes les traces sont activées par défaut.

XMS_TRACE_FILE_PATH

La variable d'environnement **XMS_TRACE_FILE_PATH** indique le nom de chemin qualifié complet du répertoire dans lequel les enregistrements de trace et FFDC sont écrits, si vous souhaitez que ces enregistrements soient écrits dans un autre emplacement à partir du répertoire de travail en cours.

XMS_TRACE_FORMAT

La variable d'environnement **XMS_TRACE_FORMAT** spécifie le format de trace requis, qui peut être BASIC ou ADVANCED.

XMS_TRACE_SPECIFICATION

La variable d'environnement **XMS_TRACE_SPECIFICATION** remplace les paramètres de trace définis dans la section [Trace d'un fichier de configuration d'application](#). **XMS_TRACE_SPECIFICATION** s'applique à IBM MQ classes for XMS .NET Framework uniquement.

Pour plus d'informations, voir [Traçage des applications XMS .NET](#) et [Traçage des applications XMS .NET à l'aide des variables d'environnement XMS](#).

Multi **Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms**

Vous pouvez modifier le comportement d' IBM MQ ou d'un gestionnaire de files d'attente individuel en fonction des besoins de votre installation en éditant les informations dans les fichiers de configuration (. ini). Vous pouvez également modifier les options de configuration pour IBM MQ MQI clients.

Pourquoi et quand exécuter cette tâche

Vous pouvez modifier les informations de configuration de IBM MQ au niveau du noeud ou du gestionnaire de files d'attente en modifiant les valeurs spécifiées dans un ensemble d'attributs (ou de paramètres) de configuration qui régissent IBM MQ.



Un fichier de configuration (ou fichier de strophes) contient une ou plusieurs strophes, qui sont des groupes de lignes du fichier . ini ayant ensemble une fonction commune ou définissant une partie d'un système, telles que des fonctions de journal, des fonctions de canal et des services installables. Vous pouvez modifier les attributs de configuration IBM MQ dans les fichiers de configuration suivants:

fichier de configuration IBM MQ , mqs . ini

Le fichier mqs . ini a des effets sur l'ensemble du noeud. Il existe un fichier mqs . ini pour chaque installation IBM MQ .

Etant donné que le fichier de configuration IBM MQ est utilisé pour localiser les données associées aux gestionnaires de files d'attente, un fichier de configuration inexistant ou incorrect peut entraîner l'échec de certaines ou de toutes les commandes MQSC. De plus, les applications ne peuvent pas se connecter à un gestionnaire de files d'attente qui n'est pas défini dans le fichier de configuration IBM MQ .

Fichier de configuration de l'installation, mqinst.ini

  Sur les systèmes AIX and Linux , le fichier de configuration de l'installation, mqinst . ini, contient des informations sur toutes les installations IBM MQ . Le fichier mqinst . ini ne doit pas être édité ou référencé directement car son format n'est pas fixe et peut changer. A la place, vous devez l'éditer à l'aide de commandes.

Fichier de configuration du gestionnaire de files d'attente, qm . ini

Les effets du fichier qm . ini sont modifiés pour des gestionnaires de files d'attente spécifiques. Il existe un fichier qm . ini pour chaque gestionnaire de files d'attente sur le noeud.

Fichier de configuration IBM MQ MQI client , mqclient.ini

Les options de configuration de IBM MQ MQI clients sont conservées séparément dans le fichier de configuration client, généralement nommé mqclient.ini.

Fichier de configuration de trace d'activité, mqat.ini


Le fichier mqat.ini est utilisé pour configurer le comportement de la trace d'activité.

Vous devrez peut-être éditer un fichier de configuration si, par exemple:

- Vous perdez un fichier de configuration. (Récupérer à partir de la sauvegarde si vous le pouvez.)
- Vous devez déplacer un ou plusieurs gestionnaires de files d'attente vers un nouveau répertoire.
- Vous devez modifier votre gestionnaire de files d'attente par défaut. Cela peut se produire si vous supprimez accidentellement le gestionnaire de files d'attente existant.
- Le support IBM vous conseille de le faire.

Important : Les modifications que vous apportez à un fichier de configuration ne prennent généralement effet qu'au prochain démarrage du gestionnaire de files d'attente.

Points à noter sur l'édition des fichiers de configuration :


- Les valeurs des attributs d'un fichier de configuration sont définies en fonction des priorités suivantes:
 - Les paramètres entrés sur la ligne de commande sont prioritaires sur les valeurs définies dans les fichiers de configuration.
 - Les valeurs définies dans les fichiers qm.ini sont prioritaires par rapport aux valeurs définies dans le fichier mqs.ini.
- Après l'installation, vous pouvez éditer les valeurs par défaut dans les fichiers de configuration IBM MQ.
- Lors de la sauvegarde d'un gestionnaire de files d'attente, n'oubliez pas d'inclure à la fois son fichier de configuration (qm.ini) et le fichier de configuration IBM MQ central (mqs.ini).
- Si vous définissez une valeur incorrecte sur un attribut de fichier de configuration, l'effet est identique à l'absence complète de l'attribut. La valeur est ignorée et un message d'opérateur est émis pour indiquer l'incident.
-  Sous IBM i, les fichiers .ini sont des fichiers STREAM résidant dans le système IFS.
- Il existe un certain nombre de règles de syntaxe pour le format du fichier mqat.ini. Pour plus d'informations, voir [Trace d'activité d'application Configuration du comportement de trace d'activité à l'aide de mqat.ini](#).



Procédure

1. Avant d'éditer un fichier de configuration, sauvegardez ce fichier afin de disposer d'une copie à laquelle vous pouvez revenir, le cas échéant.

2. Editez le fichier de configuration .ini de l'une des manières suivantes:

- Manuellement à l'aide d'un éditeur de texte standard. Les commentaires peuvent être inclus dans les fichiers de configuration en ajoutant un ";" ou un caractère "#" avant le texte du commentaire. Si vous souhaitez utiliser un ";" ou un caractère "#" sans qu'il représente un commentaire, vous pouvez le préfixer avec un caractère "\". Le caractère est ensuite utilisé dans les données de configuration.
- Automatiquement, à l'aide de commandes qui modifient la configuration des gestionnaires de files d'attente sur le noeud. Pour plus d'informations, voir [Référence des commandes](#).

 Par exemple, la Windows commande spécifique **amqmdain** met à jour automatiquement un sous-ensemble des propriétés qm.ini. Pour plus d'informations, voir [amqmdain](#).

-   Sous Linux (x86 et x86-64) et Windows, vous pouvez mettre à jour un sous-ensemble des propriétés qm.ini à l'aide de IBM MQ Explorer. Pour plus d'informations, voir [Configuration d'IBM MQ à l'aide de MQ Explorer](#).

Remarque : Etant donné que la modification des services installables et de leurs composants a des implications importantes, les services installables sont en lecture seule dans le IBM MQ Explorer. Par conséquent, vous devez apporter des modifications aux services installables en éditant le fichier `qm.ini`. Pour plus d'informations, voir «Section de service du fichier `qm.ini`», à la page 157.

Tâches associées

Administration de IBM MQ

Multi Fichier de configuration IBM MQ , `mqs.ini`

Le fichier de configuration IBM MQ , `mqs.ini`, contient des informations relatives à tous les gestionnaires de files d'attente sur le noeud. Il est créé automatiquement lors de l'installation.

Remarque : Pour plus d'informations sur la façon et le moment d'éditer le fichier `mqs.ini` et sur le moment où les modifications que vous apportez au fichier prennent effet, voir «Modification des informations de configuration IBM MQ dans les fichiers `.ini` sur Multiplatforms», à la page 96.

emplacements des répertoires

Linux **AIX** Sous AIX and Linux, le répertoire de données et le répertoire de journaux sont toujours `/var/mqm` et `/var/mqm/log` respectivement.

Windows Sur les systèmes Windows , l'emplacement du répertoire de données `mqs.iniet` l'emplacement du répertoire de journaux sont stockés dans le registre, car leur emplacement peut varier. Les informations de configuration de l'installation, qui sont contenues dans `mqinst.ini` sur les systèmes AIX and Linux , se trouvent également dans le registre, car il n'existe pas de fichier `mqinst.ini` sur Windows (voir «Fichier de configuration de l'installation, `mqinst.ini`», à la page 174).

Windows Le fichier `mqs.ini` pour les systèmes Windows est fourni par le chemin WorkPath spécifié dans la clé `HKLM\SOFTWARE\IBM\IBM MQ`. Elle comprend :

- Noms des gestionnaires de files d'attente
- Nom du gestionnaire de files d'attente par défaut
- Emplacement des fichiers associés à chacun d'eux

IBM i Sous IBM i, le fichier `mqs.ini` est stocké dans `/QIBM/UserData/mqm`. Le fichier contient:

- Noms des gestionnaires de files d'attente.
- Nom du gestionnaire de files d'attente par défaut.
- Emplacement des fichiers associés à chaque gestionnaire de files d'attente.
- Informations identifiant les exits API (voir [Configuration des exits API](#) pour plus d'informations).

En particulier, le fichier `mqs.ini` est utilisé pour localiser les données associées à chaque gestionnaire de files d'attente.

Exemple de fichier `mqs.ini` pour AIX and Linux

Linux **AIX**

```
#*****#
#* Module Name: mqs.ini                                     #*
#* Type       : IBM MQ Machine-wide Configuration File     #*
#* Function    : Define IBM MQ resources for an entire machine #*
#*****#
#* Notes      :                                           #*
#* 1) This is the installation time default configuration   #*
#*                                                    #*
#*****#
AllQueueManagers:
#*****#
#* The path to the qmgrs directory, below which queue manager data #*
```

```

## is stored ##
#####
DefaultPrefix=/var/mqm

LogDefaults:
  LogPrimaryFiles=3
  LogSecondaryFiles=2
  LogFilePages=4096
  LogType=CIRCULAR
  LogBufferPages=0
  LogDefaultPath=/var/mqm/log

QueueManager:
  Name=saturn.queue.manager
  Prefix=/var/mqm
  Directory=saturn!queue!manager
  InstallationName=Installation1

QueueManager:
  Name=pluto.queue.manager
  Prefix=/var/mqm
  Directory=pluto!queue!manager
  InstallationName=Installation2

DefaultQueueManager:
  Name=saturn.queue.manager

ApiExitTemplate:
  Name=OurPayrollQueueAuditor
  Sequence=2
  Function=EntryPoint
  Module=/usr/ABC/auditor
  Data=123

ApiExitCommon:
  Name=MQPoliceman
  Sequence=1
  Function=EntryPoint
  Module=/usr/MQPolice/tmqp
  Data=CheckEverything

```

Exemple de fichier mqs . ini pour Windows

Windows

```

#####
## Module Name: mqs.ini ##
## Type : IBM MQ Machine-wide Configuration File ##
## Function : Define IBM MQ resources for an entire machine ##
#####
## Notes : ##
## 1) This is the installation time default configuration ##
##
#####
AllQueueManagers:
#####
## The path to the qmgrs directory, below which queue manager data ##
## is stored ##
#####
DefaultPrefix=C:\ProgramData\IBM\MQ

LogDefaults:
  LogPrimaryFiles=3
  LogSecondaryFiles=2
  LogFilePages=4096
  LogType=CIRCULAR
  LogBufferPages=0
  LogDefaultPath=C:\ProgramData\IBM\MQ\log

QueueManager:
  Name=saturn.queue.manager
  Prefix=C:\ProgramData\IBM\MQ
  Directory=saturn!queue!manager
  InstallationName=Installation1

QueueManager:
  Name=pluto.queue.manager
  Prefix=C:\ProgramData\IBM\MQ

```

```

Directory=pluto!queue!manager
InstallationName=Installation2

DefaultQueueManager:
  Name=saturn.queue.manager

ApiExitTemplate:
  Name=OurPayrollQueueAuditor
  Sequence=2
  Function=EntryPoint
  Module=C:\usr\ABC\auditor
  Data=123

ApiExitCommon:
  Name=MQPoliceman
  Sequence=1
  Function=EntryPoint
  Module=C:\usr\MQPolice\tmpq
  Data=CheckEverything

```

Exemple de fichier mqs.ini pour IBM i

IBM i

```

#*****#
#* Module Name: mqs.ini                                     *#
#* Type       : IBM MQ Configuration File                 *#
#* Function   : Define IBM MQ resources for the node     *#
#*           *#
#*****#
#* Notes      :                                           *#
#* 1) This is an example IBM MQ configuration file       *#
#*           *#
#*****#
AllQueueManagers:
#*****#
#* The path to the qmgrs directory, within which queue manager data *#
#* is stored                                           *#
#*****#
DefaultPrefix=/QIBM/UserData/mqm

QueueManager:
Name=saturn.queue.manager
Prefix=/QIBM/UserData/mqm
Library=QMSATURN.Q
Directory=saturn!queue!manager

QueueManager:
Name=pluto.queue.manager
Prefix=/QIBM/UserData/mqm
Library=QMPLUTO.QU
Directory=pluto!queue!manager

DefaultQueueManager:
Name=saturn.queue.manager

```

Remarques :

1. IBM MQ sur le noeud utilise les emplacements par défaut pour les gestionnaires de files d'attente et les journaux.
2. Le gestionnaire de files d'attente saturn.queue.manager est le gestionnaire de files d'attente par défaut pour le noeud. Le répertoire des fichiers associés à ce gestionnaire de files d'attente a été automatiquement transformé en nom de fichier valide pour le système de fichiers.
3. Etant donné que le fichier de configuration IBM MQ est utilisé pour localiser les données associées aux gestionnaires de files d'attente, un fichier de configuration inexistant ou incorrect peut entraîner l'échec de certaines ou de toutes les commandes IBM MQ . De plus, les applications ne peuvent pas se connecter à un gestionnaire de files d'attente qui n'est pas défini dans le fichier de configuration IBM MQ .

mqs.ini strophes



Avertissement : Cette rubrique contient des liens vers des informations supplémentaires sur les strophes du fichier `mqs.ini`. Chaque section contient des informations sur les paramètres de cette section.

Multi Récapitulatif des sections et attributs du fichier `mqs.ini`

Récapitulatif des attributs des sections du fichier de configuration IBM MQ, `mqs.ini`, avec des liens vers des informations supplémentaires.

<i>Tableau 10. Sections du fichier <code>mqs.ini</code></i>	
Strophe et attributs	Description des attributs
Strophe des gestionnaires AllQueue	
DefaultPrefix	Chemin d'accès au répertoire <code>qmgrs</code> , dans lequel les données du gestionnaire de files d'attente sont conservées.
ALW DefaultEphemeralpar défaut	Chemin d'accès au répertoire dans lequel les données éphémères du gestionnaire de files d'attente sont conservées.
Multi ConvEBCDICNewline	Comment IBM MQ convertit le caractère NL EBCDIC au format ASCII
ApiExitStrophe commune et ApiExitStrophe de modèle	
Nom	Nom descriptif de l'exit API qui lui a été transmis dans la zone <code>ExitInfoNom</code> de la structure <code>MQXP</code> .
Fonction	Nom du point d'entrée de fonction dans le module contenant le code d'exit d'API.
Module	Module contenant le code d'exit d'API.
Données	Données à transmettre à l'exit API dans la zone <code>ExitData</code> de la structure <code>MQXP</code> .
Sequene	Séquence dans laquelle cet exit d'API est appelé par rapport à d'autres exits d'API.
DefaultQueueStrophe de gestionnaire	
Nom	Nom du gestionnaire de files d'attente qui traite les commandes pour lesquelles un nom de gestionnaire de files d'attente n'est pas explicitement spécifié.
StropheExitProperties	
CLWLMode	Indique si l'exit CLWL (cluster workloac) s'exécute en mode FAST ou SAFE.
sectionLogDefaults	
LogPrimaryFiles	Fichiers journaux alloués lors de la création du gestionnaire de files d'attente.
LogSecondaryFiles	Fichiers journaux alloués lorsque les fichiers principaux sont épuisés.
LogFilePages	Nombre de pages du fichier journal. (La taille du fichier journal est spécifiée en unités de pages de 4 ko.)

Tableau 10. Sections du fichier mqs.ini (suite)

Strophe et attributs	Description des attributs
<u>LogType</u>	Type de journalisation à utiliser par le gestionnaire de files d'attente (circulaire ou linéaire).
<u>LogBufferPages</u>	Quantité de mémoire allouée aux enregistrements de mémoire tampon pour l'écriture, en spécifiant la taille des mémoires tampon en unités de pages de 4 Ko.
<u>LogDefaultPath</u>	Répertoire dans lequel résident les fichiers journaux d'un gestionnaire de files d'attente.
<u>LogWriteIntegrity</u>	Méthode utilisée par le consignateur pour écrire des enregistrements de journal de manière fiable.
StropheQueueManager	
<u>Nom</u>	Nom du gestionnaire de files d'attente.
<u>Préfixe</u>	Emplacement de stockage des fichiers du gestionnaire de files d'attente.
<u>Répertoire</u>	Nom du sous-répertoire sous le répertoire <code>prefix\QMGRS</code> dans lequel les fichiers du gestionnaire de files d'attente sont stockés.
<u>DataPath</u>	Un chemin de données explicite fourni lors de la création du gestionnaire de files d'attente remplace le préfixe et le répertoire comme chemin d'accès aux données du gestionnaire de files d'attente.
<u>InstallationName</u>	Nom de l'installation IBM MQ associée à ce gestionnaire de files d'attente.
<u>EphemeralPrefix</u>	Emplacement de stockage des données éphémères du gestionnaire de files d'attente.

Multi **Strophe AllQueueManagers du fichier mqs.ini**

La section des gestionnaires AllQueue peut spécifier le chemin d'accès au répertoire `qmgrs` dans lequel sont stockés les fichiers associés à un gestionnaire de files d'attente, le chemin d'accès à la bibliothèque exécutable et la méthode de conversion des données au format EBCDIC au format ASCII.

Utilisez la section AllQueue du fichier `mqs.ini` pour spécifier les informations relatives à tous les gestionnaires de files d'attente.

Windows **Linux** Sous Linux (x86 et x86-64) et Windows, vous pouvez également utiliser la page de propriétés IBM MQ Explorer General et Extended IBM MQ.

DefaultPrefix= nom_répertoire

Cet attribut indique le chemin d'accès au répertoire `qmgrs`, dans lequel les données du gestionnaire de files d'attente sont conservées.

Si vous modifiez le préfixe par défaut du gestionnaire de files d'attente, répliquez la structure de répertoire créée lors de l'installation. En particulier, vous devez créer la structure `qmgrs`. Arrêtez IBM MQ avant de modifier le préfixe par défaut et redémarrez IBM MQ uniquement après avoir déplacé les structures vers le nouvel emplacement et modifié le préfixe par défaut.

Remarque : **ALW** Ne supprimez pas le répertoire `/var/mqm/errors` sur les systèmes AIX and Linux, ni le répertoire `\errors` sur les systèmes Windows.

Au lieu de changer le préfixe par défaut, vous pouvez utiliser la variable d'environnement **MQSPREFIX** pour remplacer le préfixe **DefaultPrefix** de la commande **crtmqm**.

En raison des restrictions du système d'exploitation, conservez le chemin fourni suffisamment court pour que la somme de la longueur du chemin et de tout nom de gestionnaire de files d'attente ne dépasse pas 70 caractères.

Multi **DefaultEphemeralPrefix = nom_répertoire**

Cet attribut indique le chemin d'accès au répertoire dans lequel sont conservées les données éphémères du gestionnaire de files d'attente, telles que les sockets IPC, et est utilisé uniquement pour définir le **EphemeralPrefix** d'un gestionnaire de files d'attente lors de la création d'un gestionnaire de files d'attente. En outre, vous devez créer le répertoire vous-même si vous modifiez la valeur par défaut. Vous devez créer le répertoire de données éphémère avec des droits qui permettent au groupe IBM MQ d'accéder en écriture à ce répertoire.

Au lieu de modifier le fichier `mqm.ini`, vous pouvez utiliser la variable d'environnement **MQ_EPHEMERAL_PREFIX** pour remplacer le **DefaultEphemeralPrefix** de la commande **crtmqm**.

En raison des restrictions du système d'exploitation, le préfixe éphémère par défaut est limité à :

- ▶ **Linux** ▶ **AIX** 12 caractères sur les plateformes AIX and Linux .
- ▶ **IBM i** 24 caractères sous IBM i.

▶ **MQ Appliance** **DefaultEphemeralPrefix** n'est pas pris en charge sur IBM MQ Appliance.

Multi **ConvEBCDICNewline= NL_TO_LF | TABLE | ISO**

Les pages de codes EBCDIC contiennent un caractère de retour à la ligne (NL) qui n'est pas pris en charge par les pages de codes ASCII (bien que certaines variantes ISO de l'ASCII contiennent un équivalent). Utilisez l'attribut **ConvEBCDICNewline** pour indiquer comment IBM MQ doit convertir le caractère NL EBCDIC au format ASCII.

▶ **IBM i** Sous IBM MQ for IBM i, le CCSID 1253 est considéré comme un CCSID ISO et NL_TO_LF affecte les conversions ISO et ASCII.

▶ **z/OS** L'attribut **ConvEBCDICNewline** n'est pas disponible sous z/OS. Le comportement sous z/OS est équivalent à `ConvEBCDICNewline=TABLE`. Notez que la valeur par défaut sur d'autres plateformes peut être différente.

NL_TO_LF

Convertissez le caractère NL EBCDIC (X'15') en caractère de saut de ligne ASCII, LF (X'0A'), pour toutes les conversions EBCDIC en ASCII.

NL_TO_LF est la valeur par défaut.

TABLE

Convertissez le caractère NL EBCDIC en fonction des tables de conversion utilisées sur votre plateforme pour toutes les conversions EBCDIC en ASCII.

L'effet de ce type de conversion peut varier d'une plateforme à l'autre et d'une langue à l'autre ; même sur la même plateforme, le comportement peut varier si vous utilisez des CCSID différents.

ISO

Convertir:

- CCSID ISO utilisant la méthode TABLE
- Tous les autres CCSID utilisant la méthode NL_TO_LF

Les CCSID ISO possibles sont indiqués dans la [Tableau 11](#), à la page 104.

Tableau 11. Liste des CCSID ISO possibles	
CCSID	Défini par code
819	ISO8859-1
912	ISO8859-2
915	ISO8859-5
1089	ISO8859-6
813	ISO8859-7
916	ISO8859-8
920	ISO8859-9
1051	roman8

Si le CCSID ASCII n'est pas un sous-ensemble ISO, **ConvEBCDICNewLine** prend par défaut la valeur `NL_TO_LF`.

Depuis IBM MQ 9.1.0 Fix Pack 2 et IBM MQ 9.1.2, vous pouvez utiliser la variable d'environnement **AMQ_CONVEBCDICNEWLINE** à la place de l'attribut de strophe **ConvEBCDICNewLine**, par exemple pour fournir la fonctionnalité **ConvEBCDICNewLine** côté client dans les situations où le fichier `mqs.ini` ne peut pas être utilisé. La variable d'environnement prend les mêmes valeurs (`NL_TO_LF`, `TABLEOU` ou `ISO`) que l'attribut **ConvEBCDICNewLine**. L'attribut de strophe est prioritaire si l'attribut et la variable d'environnement sont définis.

Multi **ApiExitSections de modèle communes et ApiExitdu fichier mqs.ini**

Le modèle `ApiExit` et les sections communes `ApiExit` identifient les routines d'exit d'API pour tous les gestionnaires de files d'attente.

Utilisez le modèle `ApiExit` et les sections communes `ApiExit` du fichier `mqs.ini` pour identifier les routines d'exit d'API pour tous les gestionnaires de files d'attente. (Pour identifier les routines d'exit d'API des gestionnaires de files d'attente individuels, utilisez la section locale `ApiExit`, comme décrit dans «[ApiExitStrophe locale du fichier qm.ini](#)», à la page 124.)

Windows **Linux** Vous pouvez également, sous Linux (x86 et x86-64) et Windows, utiliser la page de propriétés IBM MQ Explorer `Exits IBM MQ`.

Windows Sous Windows, vous pouvez également utiliser la commande **amqmdain** pour modifier les entrées des exits API.

Pour plus d'informations sur l'utilisation de ces attributs, voir [Configuration des exits API](#).

Name=nom_exit_pix

Nom descriptif de l'exit API qui lui a été transmis dans la zone `ExitInfoNom` de la structure `MQXP`.

Ce nom doit être unique, ne pas dépasser 48 caractères et contenir uniquement des caractères valides pour les noms d'objets IBM MQ (par exemple, les noms de file d'attente).

Function=nom_fonction

Nom du point d'entrée de fonction dans le module contenant le code d'exit d'API. Ce point d'entrée est la fonction `MQ_INIT_EXIT`.

La longueur de cette zone est limitée à `MQ_EXIT_NAME_LENGTH`.

Module=nom_module

Module contenant le code d'exit d'API.

Ce champ est utilisé tel quel s'il contient le chemin d'accès complet au module. Si cette zone contient uniquement le nom du module, le module est localisé à l'aide de l'attribut **ExitsDefaultPath** dans la strophe `ExitPath` du fichier `qm.ini`.

Sur les plateformes qui prennent en charge des bibliothèques à unités d'exécution distinctes, vous devez fournir à la fois une version non à unités d'exécution et une version à unités d'exécution du module d'exit API. La version à unités d'exécution doit avoir un suffixe `_r`. La version à unités d'exécution du module de remplacement d'application IBM MQ ajoute implicitement `_r` au nom de module donné avant son chargement.

La longueur de cette zone est limitée à la longueur de chemin maximale prise en charge par la plateforme.

Data=nom_données

Données à transmettre à l'exit API dans la zone ExitData de la structure MQAXP.

Si vous incluez cet attribut, les blancs de début et de fin sont supprimés, la chaîne restante est tronquée à 32 caractères et le résultat est transmis à l'exit. Si vous omettez cet attribut, la valeur par défaut de 32 blancs est transmise à l'exit.

La longueur maximale de cette zone est de 32 caractères.

Sequence=numéro_séquence

Séquence dans laquelle cet exit d'API est appelé par rapport à d'autres exits d'API. Un exit avec un numéro de séquence faible est appelé avant un exit avec un numéro de séquence plus élevé. Il n'est pas nécessaire que la numérotation de séquence des exits soit contiguë. Une séquence de 1, 2, 3 a le même résultat qu'une séquence de 7, 42, 1096. Si deux exits ont le même numéro de séquence, le gestionnaire de files d'attente décide lequel appeler en premier. Vous pouvez déterminer qui a été appelé après l'événement en plaçant l'heure ou un marqueur dans la zone ExitChainindiquée par ExitChainAreaPtr dans MQAXP ou en écrivant votre propre fichier journal.

Cet attribut est une valeur numérique non signée.

Multi Section DefaultQueuedu gestionnaire du fichier mqs.ini

La section DefaultQueuedu gestionnaire indique le gestionnaire de files d'attente par défaut pour le noeud.

Utilisez la section DefaultQueuedu gestionnaire dans le fichier `mqs.ini` pour spécifier le gestionnaire de files d'attente par défaut.

Windows **Linux** Vous pouvez également, sous Linux (x86 et x86-64) et Windows, utiliser la IBM MQ Explorer page de propriétés de General IBM MQ .

Nom = default_queue_manager

Le gestionnaire de files d'attente par défaut traite toutes les commandes pour lesquelles un nom de gestionnaire de files d'attente n'est pas explicitement spécifié. L'attribut **DefaultQueueManager** est automatiquement mis à jour si vous créez un gestionnaire de files d'attente par défaut. Si vous créez par inadvertance un nouveau gestionnaire de files d'attente par défaut et que vous souhaitez revenir à l'original, modifiez manuellement l'attribut **DefaultQueueManager** .

Multi Section ExitProperties du fichier mqs.ini

La section ExitProperties indique les options de configuration utilisées par les programmes d'exit du gestionnaire de files d'attente.

Utilisez la section ExitProperties du fichier `mqs.ini` pour spécifier les options de configuration utilisées par les programmes d'exit du gestionnaire de files d'attente.

Windows **Linux** Vous pouvez également, sous Linux (x86 et x86-64) et Windows, utiliser la page de propriétés IBM MQ Explorer Extended IBM MQ .

CLWLMode = SAFE (par défaut) | FAST

L'exit CLWL (Cluster Workload) vous permet de spécifier la file d'attente de cluster à ouvrir dans le cluster en réponse à un appel MQI (par exemple, MQOPEN, MQPUT). L'exit CLWL s'exécute en mode FAST ou SAFE en fonction de la valeur que vous spécifiez dans l'attribut **CLWLMode** . Si vous omettez l'attribut **CLWLMode** , l'exit de charge de travail de cluster s'exécute en mode SAFE.

SECURISEE

Exécutez l'exit CLWL dans un processus distinct du gestionnaire de files d'attente. Il s'agit de l'option par défaut.

Si un problème se produit avec l'exit CLWL écrit par l'utilisateur lors de l'exécution en mode SAFE, les événements suivants se produisent:

- Le processus serveur CLWL (amqzlw0) échoue.
- Le gestionnaire de files d'attente redémarre le processus serveur CLWL.
- L'erreur vous est signalée dans le journal des erreurs. Si un appel MQI est en cours, vous recevez une notification sous la forme d'un code retour.

L'intégrité du gestionnaire de files d'attente est préservée.

Remarque : L'exécution de l'exit CLWL dans un processus distinct peut affecter les performances.

RAPIDE

Exécutez l'exit de cluster en ligne dans le processus de gestionnaire de files d'attente.

La spécification de cette option améliore les performances en évitant les coûts de changement de processus associés à l'exécution en mode SAFE, mais au détriment de l'intégrité du gestionnaire de files d'attente. Vous ne devriez exécuter l'exit CLWL en mode FAST que si vous êtes convaincu qu'il n'y a aucun problème avec votre exit CLWL, et que vous êtes particulièrement préoccupé par les performances.

Si un problème se produit lorsque l'exit CLWL s'exécute en mode FAST, le gestionnaire de files d'attente échoue et vous risquez de compromettre l'intégrité du gestionnaire de files d'attente.

Multi

Section LogDefaults du fichier mqs.ini

La section LogDefaults indique des informations sur les valeurs de journal par défaut de tous les gestionnaires de files d'attente.

Utilisez la section LogDefaults du fichier `mqs.ini` pour spécifier des informations sur les valeurs par défaut de journal pour tous les gestionnaires de files d'attente.

Windows

Linux

Vous pouvez également, sous Linux (x86 et x86-64) et Windows, utiliser la page de propriétés IBM MQ Explorer Default log settings IBM MQ.

Si vous avez besoin d'une valeur autre que la valeur par défaut, vous devez spécifier explicitement cette valeur dans la section LogDefaults.

Si la section LogDefaults n'existe pas, les valeurs par défaut IBM MQ sont utilisées. Les attributs de journal sont utilisés comme valeurs par défaut lorsque vous créez un gestionnaire de files d'attente, mais ils peuvent être remplacés si vous spécifiez les attributs de journal dans la commande `crtmqm`. Pour plus d'informations sur cette commande, voir [crtmqm](#).

Une fois qu'un gestionnaire de files d'attente a été créé, les attributs de journal de ce gestionnaire de files d'attente sont extraits des paramètres décrits dans «Section de journal du fichier `qm.ini`», à la page 147.

Remarque : La section LogDefaults fournie pour une nouvelle installation IBM MQ ne contient pas de valeurs explicites pour les attributs. L'absence d'attribut signifie que la valeur par défaut de cette valeur est utilisée lors de la création d'un nouveau gestionnaire de files d'attente. Les valeurs par défaut de la section LogDefaults sont indiquées dans «Exemple de fichier `mqs.ini` pour AIX and Linux», à la page 98 et «Exemple de fichier `mqs.ini` pour Windows», à la page 99. La valeur zéro pour l'attribut LogBufferPages signifie 512.

Le préfixe par défaut, qui est spécifié dans le fichier «Strophe AllQueueManagers du fichier `mqs.ini`», à la page 102, et le chemin d'accès au journal spécifié pour le gestionnaire de files d'attente spécifique, qui est spécifié dans le fichier «Section de journal du fichier `qm.ini`», à la page 147, permettent au gestionnaire de files d'attente et à son journal d'être sur des unités physiques différentes. Il s'agit de la méthode recommandée, bien que par défaut ils se trouvent sur la même unité.

Pour plus d'informations sur le calcul des tailles de journal, voir [«Calcul de la taille du journal»](#), à la page 710.

Remarque : Les limites indiquées dans la liste de paramètres suivante sont des limites définies par IBM MQ. Les limites du système d'exploitation peuvent réduire la taille de journal maximale possible.

LogPrimaryFiles = 3 (par défaut) | 2-254 (Windows) | 2-510 (AIX and Linux)

Fichiers journaux alloués lors de la création du gestionnaire de files d'attente.

Le nombre minimal de fichiers journaux principaux que vous pouvez avoir est 2 et le nombre maximal est 254 sous Windows ou 510 sous AIX and Linux. La valeur par défaut est 3.

Le nombre total de fichiers journaux primaires et secondaires ne doit pas dépasser 255 sous Windows ou 511 sous AIX and Linux et ne doit pas être inférieur à 3.

La valeur est examinée lorsque le gestionnaire de files d'attente est créé ou démarré. Vous pouvez le modifier après la création du gestionnaire de files d'attente. Toutefois, une modification de la valeur n'est pas effective tant que le gestionnaire de files d'attente n'est pas redémarré et que l'effet n'est pas immédiat.

LogSecondaryFiles = 2 (par défaut) | 1-253 (Windows) | 1-509 (AIX and Linux)

Fichiers journaux alloués lorsque les fichiers principaux sont épuisés.

Le nombre minimal de fichiers journaux secondaires est 1 et le nombre maximal est 253 sur Windows ou 509 sur AIX and Linux. Le nombre par défaut est 2.

Le nombre total de fichiers journaux primaires et secondaires ne doit pas dépasser 255 sous Windows ou 511 sous AIX and Linux et ne doit pas être inférieur à 3.

La valeur est examinée lorsque le gestionnaire de files d'attente est démarré. Vous pouvez modifier cette valeur, mais les modifications ne prennent effet que lorsque le gestionnaire de files d'attente est redémarré, et même dans ce cas, l'effet peut ne pas être immédiat.

LogFilePages = nombre

Les données de journal sont consignées dans une série de fichiers appelée fichiers journaux. La taille de ces fichiers est définie en unité de pages de 4 ko.

Le nombre de pages par défaut d'un fichier journal est égal à 4 096, ce qui signifie que la taille du fichier journal est égale à 16 Mo.

Sous AIX and Linux, le nombre minimal de pages de fichier journal est de 64 et sous Windows, le nombre minimal de pages de fichier journal est de 32 ; dans les deux cas, le nombre maximal est de 65 535.

Remarque : La taille des fichiers journaux spécifiée lors de la création du gestionnaire de files d'attente ne peut pas être modifiée pour un gestionnaire de files d'attente.

LogType = CIRCULAIRE (par défaut) | LINEAR

Type de journal à utiliser. La valeur par défaut est CIRCULAIRE.

REUTILISATION

Démarrez la reprise par redémarrage à l'aide du journal pour annuler les transactions qui étaient en cours lorsque le système s'est arrêté.

Pour plus d'informations sur la journalisation avec réutilisation automatique des journaux, voir [«Types de journalisation»](#), à la page 704 .

LINEAR

Pour la reprise par redémarrage et la reprise sur support ou la reprise en aval (création de données perdues ou endommagées en rélisant le contenu du journal).

Voir [«Types de journalisation»](#), à la page 704 pour une explication plus complète de la journalisation linéaire.

Si vous souhaitez modifier la valeur par défaut, vous pouvez éditer l'attribut LogType ou spécifier la journalisation linéaire à l'aide de la commande **crtmqm** .

Vous pouvez modifier la méthode de consignation après la création d'un gestionnaire de files d'attente. Pour plus d'informations, voir [migmqlog](#).

LogBufferPages=0 (valeur par défaut) |0-4096

Quantité de mémoire allouée aux enregistrements de mémoire tampon pour l'écriture, en spécifiant la taille des mémoires tampon en unités de pages de 4 Ko.

Le nombre minimal de pages tampon est 18 et le nombre maximal est 4096. Les mémoires tampon de taille plus importante améliorent le rendement, en particulier pour les messages plus grands.




Si vous indiquez 0 (valeur par défaut), le gestionnaire de files d'attente sélectionne la taille 512 (2048 Ko).

Si vous indiquez un nombre compris entre 1 et 17, la valeur par défaut du gestionnaire de files d'attente est 18 (72 Ko). Si vous indiquez un nombre compris entre 18 et 4096, le gestionnaire de files d'attente utilise le nombre spécifié pour définir la mémoire allouée.

LogDefaultPath = nom_répertoire

Répertoire dans lequel résident les fichiers journaux d'un gestionnaire de files d'attente. Le répertoire se trouve sur une unité locale sur laquelle le gestionnaire de files d'attente peut écrire et, de préférence, sur une unité différente des files d'attente de messages. La spécification d'une autre unité offre une protection supplémentaire en cas de défaillance du système.

La valeur par défaut est la suivante :

-  *DefaultPrefix*\log for IBM MQ for Windows où *DefaultPrefix* est la valeur spécifiée dans l'attribut *DefaultPrefix* de la page de propriétés All Queue Managers IBM MQ . Cette valeur est définie lors de l'installation.
-   /var/mqm/log pour les systèmes AIX and Linux .

Vous pouvez également spécifier le nom d'un répertoire dans la commande **crtmqm** à l'aide de l'indicateur **-ld** . Lorsqu'un gestionnaire de files d'attente est créé, un répertoire est également créé sous le répertoire du gestionnaire de files d'attente, qui est utilisé pour stocker les fichiers journaux. Le nom de ce répertoire est basé sur le nom du gestionnaire de files d'attente. Cela permet de s'assurer que le chemin d'accès au fichier journal est unique et qu'il respecte toutes les limitations relatives à la longueur des noms de répertoire.

Si vous n'indiquez pas **-ld** dans la commande **crtmqm** , la valeur de l'attribut **LogDefaultPath** dans le fichier `mqsc.ini` est utilisée.

Le nom du gestionnaire de files d'attente est ajouté au nom de répertoire pour garantir que plusieurs gestionnaires de files d'attente utilisent des répertoires de journaux différents.

Lorsque le gestionnaire de files d'attente est créé, une valeur **LogPath** est créée dans les attributs de journal des informations de configuration, en indiquant le nom de répertoire complet du journal du gestionnaire de files d'attente. Cette valeur permet de localiser le journal lorsque le gestionnaire de files d'attente est démarré ou supprimé.

LogWriteIntegrity =SingleWrite|DoubleWrite|TripleWrite (valeur par défaut)

Méthode utilisée par le consignateur pour écrire des enregistrements de journal de manière fiable.

TripleWrite (par défaut)

Sachez que vous pouvez sélectionner **DoubleWrite** mais ceci est interprété par le système comme étant **TripleWrite**.

SingleWrite

Vous ne devez utiliser **SingleWrite** que si le système de fichiers et l'unité hébergeant le journal de reprise IBM MQ garantissent explicitement l'atomicité des écritures 4KB .

En cas d'échec de l'écriture d'une page de 4 ko, quelle qu'en soit la raison, les deux seuls états possibles sont l'image d'avant et l'image d'après. Aucun état intermédiaire n'est possible.

Remarque : S'il y a suffisamment d'accès concurrents dans votre charge de travail persistante, la définition d'une valeur autre que la valeur par défaut, `TripleWrite`, présente un avantage potentiel minimal.

Pour plus d'informations, voir «[LogWriteIntegrity-à l'aide de SingleWrite ou de TripleWrite](#)», à la page 150.

Multi

Section QueueManager du fichier `mqs.ini`

La section QueueManager indique l'emplacement du répertoire du gestionnaire de files d'attente.

Il existe une section QueueManager pour chaque gestionnaire de files d'attente. Les attributs de cette section indiquent le nom du gestionnaire de files d'attente et le nom du répertoire contenant les fichiers associés à ce gestionnaire de files d'attente. Le nom du répertoire est basé sur le nom du gestionnaire de files d'attente, mais il est transformé si le nom du gestionnaire de files d'attente n'est pas un nom de fichier valide. Pour plus d'informations sur la transformation de nom, voir [Présentation des noms de fichier IBM MQ](#).

Name = *nom_gestionnaire_files_attente*

Nom du gestionnaire de files d'attente.

Prefix = *préfixe*

Emplacement de stockage des fichiers du gestionnaire de files d'attente. Par défaut, cette valeur est identique à la valeur spécifiée dans l'attribut **DefaultPrefix** de la section [Tous les gestionnaires de files d'attente](#) du fichier `mqs.ini`.

Directory = *nom*

Nom du sous-répertoire sous le répertoire `prefix\QMGRS` dans lequel les fichiers du gestionnaire de files d'attente sont stockés. Ce nom est basé sur le nom du gestionnaire de files d'attente, mais peut être transformé s'il existe un nom en double ou si le nom du gestionnaire de files d'attente n'est pas un nom de fichier valide.

DataPath= *chemin*

Un chemin de données explicite fourni lors de la création du gestionnaire de files d'attente remplace **Prefix** et **Directory** comme chemin d'accès aux données du gestionnaire de files d'attente.

InstallationName= *nom*

Nom de l'installation IBM MQ associée à ce gestionnaire de files d'attente. Les commandes de cette installation doivent être utilisées lors de l'interaction avec ce gestionnaire de files d'attente.

IBM i

Library = *nom*

Nom de la bibliothèque dans laquelle sont stockés les objets IBM i pertinents pour ce gestionnaire de files d'attente, par exemple les journaux et les récepteurs de journal. Ce nom est basé sur le nom du gestionnaire de files d'attente, mais peut être transformé s'il existe un nom en double ou si le nom du gestionnaire de files d'attente n'est pas un nom de bibliothèque valide.

EphemeralPrefix= *nom*

Emplacement de stockage des données éphémères du gestionnaire de files d'attente.

Par défaut, cette valeur n'est pas présente, ce qui signifie que les données sont stockées sous l'emplacement du préfixe.

La valeur est définie à partir de la valeur de la variable d'environnement **MQ_EPHEMERAL_PREFIX** ou de l'attribut **DefaultEphemeralPrefix** de la section [AllQueueManagers](#) dans le fichier `mqs.ini`, lorsque le gestionnaire de files d'attente est créé.

IBM i

En raison des restrictions du système d'exploitation, le préfixe éphémère par défaut est limité à 24 caractères sous IBM i.

Tâches associées

«[Association d'un gestionnaire de files d'attente à une installation](#)», à la page 502

Lorsque vous créez un gestionnaire de files d'attente, il est automatiquement associé à l'installation qui a émis la commande **crtmqm**. Sous AIX, Linux, and Windows, vous pouvez modifier l'installation associée à un gestionnaire de files d'attente à l'aide de la commande **setmqm**.

Windows Configuration avancée et interface d'alimentation (ACPI)

Windows prend en charge la norme ACPI (Advanced Configuration and Power Interface). Cela permet aux utilisateurs Windows dont le matériel est activé pour ACPI d'arrêter et de redémarrer les canaux lorsque le système passe en mode d'interruption et redémarre.

Utilisez la page de propriétés ACPI IBM MQ de IBM MQ Explorer pour spécifier le comportement de IBM MQ lorsque le système reçoit une demande d'interruption.

Notez que les paramètres spécifiés dans la page de propriétés ACPI IBM MQ sont appliqués uniquement lorsque le moniteur d'alertes est en cours d'exécution. L'icône Moniteur d'alertes est présente dans la barre des tâches si le moniteur d'alertes est en cours d'exécution.

DoDialog= Y | N

Affiche la boîte de dialogue au moment d'une demande d'interruption.

DenySuspend= Y | N

Refuse la demande d'interruption. Elle est utilisée si DoDialog= N, ou si DoDialog= Y et qu'une boîte de dialogue ne peut pas être affichée, par exemple, parce que le couvercle de votre bloc-notes est fermé.

CheckChannelsExécution = Y | N

Vérifie si des canaux sont en cours d'exécution. Le résultat peut déterminer le résultat des autres paramètres.

Le tableau suivant décrit l'effet de chaque combinaison de ces paramètres:

DoDialog	DenySuspend	CheckChannels en cours d'exécution	Action
N	N	N	Acceptez la demande de suspension.
N	N	Y	Acceptez la demande de suspension.
N	Y	N	Refusez la demande d'interruption.
N	Y	Y	Si des canaux sont en cours d'exécution, la demande de suspension est rejetée ; si elle n'est pas acceptée, la demande est rejetée.
Y	N	N	Afficher la boîte de dialogue (voir Remarque ; accepter la demande de suspension). Il s'agit de l'option par défaut.
Y	N	Y	Si aucun canal n'est en cours d'exécution, acceptez la demande d'interruption ; si tel est le cas, affichez la boîte de dialogue (voir Remarque ; accepter la demande).
Y	Y	N	Afficher la boîte de dialogue (Remarque ; refuser la demande d'interruption).
Y	Y	Y	Si aucun canal n'est en cours d'exécution, acceptez la demande de suspension ; si tel est le cas, affichez la boîte de dialogue (Remarque ; refuser la demande).

Remarque : Dans les cas où l'action consiste à afficher la boîte de dialogue, si la boîte de dialogue ne peut pas être affichée (par exemple, parce que le couvercle de votre bloc-notes est fermé), l'option DenySuspend est utilisée pour déterminer si la demande d'interruption est acceptée ou refusée.

Fichiers de configuration du gestionnaire de files d'attente, qm.ini

Un fichier de configuration de gestionnaire de files d'attente, `qm.ini`, contient des informations relatives à un gestionnaire de files d'attente spécifique. Les attributs que vous pouvez utiliser pour modifier la configuration d'un gestionnaire de files d'attente individuel remplacent tous les paramètres d'IBM MQ.

Il existe un fichier de configuration de gestionnaire de files d'attente pour chaque gestionnaire de files d'attente. Le fichier `qm.ini` est automatiquement créé lorsque le gestionnaire de files d'attente auquel il est associé est créé.

Remarque : Pour plus d'informations sur la façon et le moment d'éditer un fichier `qm.ini` et sur le moment où les modifications que vous apportez au fichier prennent effet, voir «[Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms](#)», à la page 96.

La commande `strmqm` vérifie la syntaxe des sections CHANNELS et SSL dans le fichier `qm.ini` avant de démarrer complètement le gestionnaire de files d'attente, ce qui permet de voir plus facilement ce qui ne va pas et de le corriger rapidement si `strmqm` détecte que le fichier `qm.ini` contient des erreurs. Pour plus d'informations, voir [strmqm](#).

Emplacement des fichiers qm.ini

Linux

AIX

Sur les systèmes AIX and Linux, un fichier `qm.ini` se trouve à la racine de l'arborescence de répertoires occupée par le gestionnaire de files d'attente. Par exemple, le chemin et le nom d'un fichier de configuration pour un gestionnaire de files d'attente appelé `QMNAME` sont les suivants:

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Windows

Sur les systèmes Windows, l'emplacement du fichier `qm.ini` est indiqué par le chemin `WorkPath` spécifié dans la clé `HKLM\SOFTWARE\IBM\WebSphere MQ`. Par exemple, le chemin et le nom d'un fichier de configuration pour un gestionnaire de files d'attente appelé `QMNAME` sont les suivants:

```
C:\ProgramData\IBM\MQ\qmgrs\QMNAME\qm.ini
```

IBM i

Un fichier `qm.ini` est stocké dans `mqmdata directory/QMNAME/qm.ini`, où `mqmdata directory` est `/QIBM/UserData/mqm` par défaut et `QMNAME` est le nom du gestionnaire de files d'attente auquel s'applique le fichier d'initialisation.

Remarque : Vous pouvez modifier le fichier `mqmdata directory` dans le fichier `mqs.ini`.

Le nom du gestionnaire de files d'attente peut comporter jusqu'à 48 caractères. Toutefois, cela ne garantit pas que le nom est valide ou unique. Par conséquent, un nom de répertoire est généré en fonction du nom du gestionnaire de files d'attente. Ce processus est appelé *transformation de nom*. Pour une description, voir [IBM MQ file names](#) et [Object names on IBM i](#).

qm.ini strophes



Avertissement :

- Cette rubrique contient des liens vers des informations supplémentaires sur les strophes du fichier `qm.ini`. Chaque section contient des informations sur les paramètres de cette section, y compris un exemple, le cas échéant.
- Chaque section indique la ou les plateformes de IBM MQ for Multiplatforms auxquelles elle s'applique.

Configuration automatique de qm.ini au démarrage

Vous pouvez configurer votre gestionnaire de files d'attente pour qu'il applique automatiquement le contenu d'un fichier ou d'un ensemble de fichiers contenant des substitutions qm.ini à chaque démarrage du gestionnaire de files d'attente.

Vous pouvez l'utiliser pour avoir une configuration qui peut être modifiée et automatiquement réexécutée lors du prochain redémarrage du gestionnaire de files d'attente. Par exemple, si les substitutions qm.ini se trouvent sur une unité montée, il est possible de disposer d'une configuration centralisée dans laquelle la version la plus récente est appliquée à chaque gestionnaire de files d'attente au fur et à mesure de leur démarrage.

Vous pouvez utiliser cette fonctionnalité pour simplifier la création d'un cluster uniforme, à l'aide de la fonctionnalité de cluster automatique. Pour un exemple, voir [«Création d'un nouveau cluster uniforme»](#), à la page 461.

Remarque : Ces substitutions ne sont appliquées qu'au démarrage du gestionnaire de files d'attente et ne peuvent pas influencer la création du gestionnaire de files d'attente. Par exemple, vous ne pouvez pas définir le nombre de fichiers journaux principaux avec cette fonction.

Avant de commencer

Vous pouvez utiliser :

1. Un fichier unique et créez un fichier texte contenant les modifications apportées au fichier qm.ini .
2. Un ensemble de fichiers au format qm.ini :
 - Pour identifier un répertoire dans lequel les configurations existeront, et
 - Dans ce répertoire, créez des fichiers, chacun avec l'extension .ini, par exemple, qminisettings.ini.

Le ou les fichiers doivent uniquement contenir la section et les paramètres **attribute=value** pour les éléments qui changent. Par exemple, pour mettre à jour l'attribut **MaxChannels** dans la strophe Channels, le fichier peut contenir:

```
Channels:  
MaxChannels=1234
```

Notez que dans les fichiers de substitution qm.ini, toute ligne préfixée avec # est traitée comme un commentaire.

Activation de la configuration automatique des attributs du fichier qm.ini

Vous pouvez configurer un nouveau gestionnaire de files d'attente en utilisant l'indicateur **-ii** dans la commande **crtmqm** et en pointant vers un fichier ou un répertoire spécifique. La valeur fournie est stockée dans le fichier qm.ini sous la section **AutoConfig**, en tant qu'attribut **IniConfig**.

Vous pouvez configurer un gestionnaire de files d'attente existant pour activer la configuration MQSC automatique en ajoutant l'attribut de section **AutoConfig IniConfig**, qui pointe vers un fichier ou un répertoire valide. Exemple :

```
AutoConfig:  
IniConfig=C:\MQ_Configuration\uniclus.ini
```

Comment la configuration automatique fonctionne-t-elle?

Lors du démarrage du gestionnaire de files d'attente, la configuration identifiée par l'attribut de strophe AutoConfig **IniConfig** est validée pour garantir la validité de la syntaxe, puis stockée dans l'arborescence de données du gestionnaire de files d'attente dans le répertoire **autocfg** sous la forme d'un fichier **cached.ini** unique.

Lorsque plusieurs fichiers d'un répertoire sont traités, ils sont traités par ordre alphabétique.

Lors du premier démarrage du gestionnaire de files d'attente, l'impossibilité de lire le fichier ou le répertoire empêche le démarrage du gestionnaire de files d'attente, avec un message d'erreur approprié à la fois sur la console et dans le journal des erreurs du gestionnaire de files d'attente.

Lors des redémarrages suivants, si le fichier ou le répertoire désigné est illisible, le fichier précédemment mis en cache est utilisé et un message écrit dans le journal des erreurs du gestionnaire de files d'attente le met en évidence.

Lorsque vous utilisez la commande **strmqm**, le contenu du fichier `cached.ini` est appliqué au fichier `qm.ini` en tant que substitutions avant que le gestionnaire de files d'attente ne soit appelé.

Cela signifie que pour un gestionnaire de files d'attente de secours, les paramètres sont lus lorsque la commande **strmqm** est traitée et non lorsque le gestionnaire de files d'attente devient actif.

Comment le fichier `qm.ini` de remplacement est-il généré?

La première fois que la configuration de l'initialisation automatique est configurée et que le gestionnaire de files d'attente démarre, une copie du fichier `qm.ini` en cours est copiée dans le sous-répertoire `autoconfig` du répertoire de données du gestionnaire de files d'attente en tant que `base_qm.ini`. Cette valeur est considérée comme la valeur de référence à partir de cette date.

A chaque démarrage du gestionnaire de files d'attente, c'est-à-dire à l'heure **strmqm**, le fichier `qm.ini` actuellement actif est supprimé et remplacé par une copie du fichier `base_qm.ini`. Ensuite, la configuration du fichier `cached.ini` est appliquée à ce fichier.

Une fois qu'un gestionnaire de files d'attente est soumis au contrôle de configuration automatique, toutes les modifications apportées au fichier `qm.ini` doivent être effectuées via le ou les fichiers qui sont indiqués à l'aide de l'attribut **IniConfig** dans la section `AutoConfig`.

Etant donné que le fichier `qm.ini` existant est supprimé au démarrage du gestionnaire de files d'attente, seule la configuration du fichier `qm.ini` fourni à l'aide de l'attribut **IniConfig** est appliquée à la ligne de base du gestionnaire de files d'attente.

Si une section ou un attribut a été modifié via la configuration d'initialisation automatique lors de démarrages précédents du gestionnaire de files d'attente, ces modifications sont supprimées sauf si elles sont toujours identifiées dans le ou les fichiers identifiés par l'attribut **IniConfig**.

En raison de la recréation du fichier `qm.ini` au démarrage du gestionnaire de files d'attente, les modifications manuelles apportées au fichier `qm.ini` sont perdues. Si vous devez vraiment rendre une modification permanente et que vous ne pouvez pas utiliser l'attribut **IniConfig** pour effectuer cette modification, vous pouvez effectuer l'une des opérations suivantes:

- Modifiez le fichier `base_qm.ini` lui-même.
- Supprimez le fichier `base_qm.ini`.

Si vous supprimez ce fichier, `base_qm.ini` est recréé au prochain démarrage du gestionnaire de files d'attente, en fonction du contenu en cours du fichier `qm.ini`. Cela *durcit* toutes les modifications en cours au fur et à mesure que la nouvelle version de référence pour les démarrages ultérieurs.

Concepts associés

[«Récapitulatif des strophes et des attributs du fichier `qm.ini`», à la page 113](#)

Récapitulatif des attributs des sections du fichier de configuration du gestionnaire de files d'attente, `qmi.ini`, avec des liens vers des informations supplémentaires.

Multi Récapitulatif des strophes et des attributs du fichier `qm.ini`

Récapitulatif des attributs des sections du fichier de configuration du gestionnaire de files d'attente, `qmi.ini`, avec des liens vers des informations supplémentaires.

Tableau 12. Sections du fichier *qm.ini*

Strophe et attributs	Description des attributs
Windows StropheAccessMode	
Windows <u>groupe d'accès</u> ¹	Un groupe de sécurité Windows dont les membres auront un accès complet à tous les fichiers de données du gestionnaire de files d'attente.
ApiExitSection locale	
<u>Nom</u>	Nom descriptif de l'exit API qui lui a été transmis dans la zone ExitInfoNom de la structure MQXP.
<u>Fonction</u>	Nom du point d'entrée de fonction dans le module contenant le code d'exit d'API.
<u>Module</u>	Module contenant le code d'exit d'API.
<u>Données</u>	Données à transmettre à l'exit API dans la zone ExitData de la structure MQXP.
<u>Sequence</u>	Séquence dans laquelle cet exit d'API est appelé par rapport à d'autres exits d'API.
V 9.4.0 Linux AIX StropheAuthToken	
<u>KeyStore</u>	Chemin d'accès au magasin de clés qui contient les certificats de clé publique de l'émetteur de confiance ou les clés symétriques.
<u>KeyStorePwdFile</u>	Chemin d'accès au fichier qui contient le mot de passe chiffré pour le magasin de clés.
<u>CertLabel</u>	Libellé de certificat d'un certificat de clé publique ou d'une clé symétrique dans le magasin de clés utilisé pour valider les jetons d'authentification.
<u>UserClaim</u>	Réclamation dans le jeton qui contient les informations d'identité de l'utilisateur que le gestionnaire de files d'attente peut adopter pour les vérifications d'autorisation.
<u>AllowOSGroups</u>	Cet attribut détermine si l'appartenance au groupe de l'utilisateur adopté est vérifiée ou non.
sectionAutoCluster	
<u>Type</u>	Type de cluster automatique. La seule option valide est Uniform, qui représente un cluster uniforme.
<u>ClusterName</u>	Nom du cluster automatique.
<u>RepositoryName1</u>	Nom du gestionnaire de files d'attente pour le premier référentiel complet dans le cluster automatique.
<u>Repository1Conname</u>	Valeur du nom de connexion (CONNNAME) indiquant comment les membres du cluster automatique doivent se connecter au gestionnaire de files d'attente.
<u>RepositoryName2</u>	Nom du gestionnaire de files d'attente pour le deuxième référentiel complet dans le cluster automatique.

Tableau 12. Sections du fichier *qm.ini* (suite)

Strophe et attributs	Description des attributs
<u>Repository2Conname</u>	Valeur du nom de connexion (CONNAME) indiquant comment les membres du cluster automatique doivent se connecter au gestionnaire de files d'attente.
StropheAutoConfig	
<u>MQSCConfig</u>	Chemin d'accès complet au fichier ou chemin d'accès à un répertoire dans lequel tous les fichiers *.mqsc sont appliqués au gestionnaire de files d'attente à chaque démarrage du gestionnaire de files d'attente.
<u>IniConfig</u>	Chemin d'accès complet au fichier ou chemin d'accès à un répertoire dans lequel tous les fichiers *.ini sont appliqués au fichier <i>qm.ini</i> à chaque démarrage du gestionnaire de files d'attente.
Strophe channels	
<u>MaxChannels</u>	Nombre maximal de canaux en cours autorisés.
<u>MaxActiveChannels</u>	Nombre maximal de canaux pouvant être actifs à tout moment.
<u>MaxInitiators</u>	Nombre maximal d'initiateurs.
<u>MQIBindType</u>	Liaison pour les applications.
<u>PipeLineLength</u>	Nombre maximal d'unités d'exécution simultanées qu'un canal utilisera.
<u>AdoptNewMCA</u>	Les types de canaux pour lesquels l'instance de canal existante peut être arrêtée afin qu'une nouvelle instance de canal puisse démarrer lorsque IBM MQ reçoit une demande de démarrage d'un canal, mais détecte qu'une instance du canal est déjà en cours d'exécution.
<u>AdoptNewMCATimeout</u>	Durée, en secondes, pendant laquelle la nouvelle instance de canal attend l'arrêt de l'ancienne instance de canal.
<u>AdoptNewMCACheck</u>	Type de vérification requis lors de l'activation de l'attribut AdoptNewMCA .
<u>ChlauthEarly-Adopter</u>	Ordre dans lequel les règles d'authentification de connexion et d'authentification de canal sont traitées.
<u>PasswordProtection</u>	Indique si les données d'identification spécifiées par une application doivent être protégées par la protection par mot de passe MQCSP, si le canal n'utilise pas le chiffrement TLS.
<u>IgnoreSeqNumberMismatch</u>	Contrôle la façon dont le gestionnaire de files d'attente traite une non-concordance de numéro de séquence lors du démarrage du canal.
Strophe de connexion	
<u>TypeDefaultBind</u>	Indique si les applications et le gestionnaire de files d'attente, qui s'exécutent dans des processus distincts, partagent certaines ressources ou pas de ressources entre elles.

Tableau 12. Sections du fichier *qm.ini* (suite)












Strophe et attributs	Description des attributs
sectionDiagnosticMessages	
<u>name</u>	Nom d'une strophe.
<u>Service</u>	Service en cours d'activation par cette section.
<u>ExcludeMessage</u>	Messages qui ne doivent pas être écrits dans le journal des erreurs du gestionnaire de files d'attente.
<u>SuppressMessage</u>	Les messages qui doivent être écrits dans le journal des erreurs du gestionnaire de files d'attente une seule fois dans un intervalle de temps spécifié.
 <u>SuppressInterval</u>	Intervalle de temps, en secondes, pendant lequel les messages spécifiés dans SuppressMessage sont écrits une seule fois dans le journal des erreurs du gestionnaire de files d'attente.
<u>Gravités</u>	Liste de niveaux de gravité séparés par des virgules.
<u>FilePath</u>	Chemin dans lequel les fichiers journaux sont écrits. (Uniquement pris en charge lorsque l'attribut Service est défini sur Fichier.)
<u>FilePrefix</u>	Préfixe des fichiers journaux. (Uniquement pris en charge lorsque l'attribut Service est défini sur Fichier.)
<u>FileSize</u>	Taille à laquelle le journal est repassé. (Uniquement pris en charge lorsque l'attribut Service est défini sur Fichier.)
<u>Format</u>	Format du fichier. (Uniquement pris en charge lorsque l'attribut Service est défini sur Fichier.)
  <u>Syslog</u>	Service Syslog qui envoie tous les messages non filtrés à syslog à l'aide de la spécification de messages de diagnostic au <u>format JSON</u> .
  <u>Ident</u>	Valeur ident associée aux entrées syslog. (Pris en charge uniquement lorsque l'attribut Service est défini sur Syslog.)
StropheExitPath	
<u>ExitsDefaultPath</u>	Chemin d'accès aux programmes d'exit utilisateur sur le système de gestionnaire de files d'attente (32 bits).
<u>ExitsDefaultPath64</u>	Chemin d'accès aux programmes d'exit utilisateur sur le système de gestionnaire de files d'attente (64 bits).
ExitPropertiesStrophe locale	
<u>CLWLMode</u>	Indique si l'exit CLWL (cluster workloac) s'exécute en mode FAST ou SAFE.
   Strophe de système de fichiers	
   <u>ValidateAuth</u>	Autorisez les utilisateurs qui ne sont pas membres du groupe mqm à accéder aux répertoires et fichiers d'erreurs.
Strophe de journal	

Tableau 12. Sections du fichier *qm.ini* (suite)

Strophe et attributs	Description des attributs
LogPrimaryFiles	Fichiers journaux alloués lors de la création du gestionnaire de files d'attente.
LogSecondaryFiles	Fichiers journaux alloués lorsque les fichiers principaux sont épuisés.
LogFilePages	Nombre de pages du fichier journal. (La taille du fichier journal est spécifiée en unités de pages de 4 ko.)
LogType	Type de journalisation à utiliser par le gestionnaire de files d'attente (circulaire ou linéaire).
LogBufferPages	Quantité de mémoire allouée aux enregistrements de mémoire tampon pour l'écriture, en spécifiant la taille des mémoires tampon en unités de pages de 4 Ko.
LogPath	Répertoire dans lequel résident les fichiers journaux d'un gestionnaire de files d'attente.
LogWriteIntegrity	Méthode utilisée par le consignateur pour écrire des enregistrements de journal de manière fiable.
LogManagement	Méthode utilisée pour gérer les extensions de journal, manuellement ou par le gestionnaire de files d'attente.
Windows StropheLU62	
Windows NomTP	Nom du programme transactionnel à démarrer sur le site distant.
Windows Library1	Nom de la DLL APPC.
Windows Library2	Identique à Library1, utilisé si le code est stocké dans deux bibliothèques distinctes.
CP4I StropheNativeHAInstance	
«Nom» , à la page 151	Nom d'instance utilisé lors de la création de l'instance de gestionnaire de files d'attente.
«ReplicationAddress» , à la page 151	Nom d'hôte, IPv4 décimale à point ou adresse au format hexadécimal IPv6 de l'instance.
CP4I NativeHALocal	
«LocalName» , à la page 152	Nom de la section d'instance NativeHALocal, issu du nom d'instance de réplique de journal spécifié lors de la création du gestionnaire de files d'attente Native HA.
«KeyRepository» , à la page 152	Emplacement du référentiel de clés qui contient le certificat numérique à utiliser pour la protection du trafic de réplification des journaux.
«CertificateLabel» , à la page 153	Libellé de certificat identifiant le certificat numérique à utiliser pour la protection du trafic de réplification des journaux.
«CipherSpec» , à la page 153	Le MQ CipherSpec à utiliser pour protéger le trafic de réplification des journaux.

Tableau 12. Sections du fichier *qm.ini* (suite)

Strophe et attributs	Description des attributs
« LocalAddress », à la page 153	Adresse de l'interface réseau locale qui accepte le trafic de réplication de journal.
« HeartbeatInterval », à la page 153	L'intervalle des pulsations définit la fréquence en millisecondes à laquelle une instance active d'un gestionnaire de files d'attente Native HA envoie une pulsation réseau.
« HeartbeatTimeout », à la page 153	Le dépassement du délai d'attente du signal de présence définit le temps pendant lequel une réplique d'instance d'un gestionnaire de files d'attente Native HA attend avant de considérer que l'instance active ne répondra pas.
« RetryInterval », à la page 153	L'intervalle entre les nouvelles tentatives définit la fréquence en millisecondes à laquelle un gestionnaire de files d'attente Native HA doit retenter un lien de réplication défectueux.
Windows Strophe NETBIOS	
Windows LocalName	Nom sous lequel cette machine est connue sur le réseau local.
Windows AdapterNum	Numéro de la carte de réseau local.
Windows NumSess	Nombre de sessions à allouer.
Windows NumCmds	Nombre de commandes à allouer.
Windows NumNames	Nombre de noms à allouer.
Windows Library1	Nom de la DLL NetBIOS .
stropheQMErrorLog	
ErrorLog	Indique la taille du journal des erreurs du gestionnaire de files d'attente qui est copié dans la sauvegarde.
ExcludeMessage	Indique les messages qui ne doivent pas être écrits dans le journal des erreurs du gestionnaire de files d'attente.
SuppressMessage	Indique les messages qui sont écrits dans le journal des erreurs du gestionnaire de files d'attente une seule fois dans un intervalle de temps spécifié.
SuppressInterval	Indique l'intervalle de temps, en secondes, pendant lequel les messages spécifiés dans SuppressMessage sont écrits une seule fois dans le journal des erreurs du gestionnaire de files d'attente.
Linux AIX Strophe Mode restreint ²	
Linux AIX ApplicationGroup	Nom de la file d'attente de transmission locale dans laquelle les messages éloignés sont insérés si une file d'attente de transmission n'est pas explicitement définie pour leur destination.
Strophe de sécurité	

Tableau 12. Sections du fichier qm.ini (suite)









Strophe et attributs	Description des attributs
<u>ClusterQueueAccessControl</u>	Vérifiez le contrôle d'accès des files d'attente de cluster ou des files d'attente complètes hébergées sur les gestionnaires de files d'attente de cluster.
 <u>GroupModel</u>	Indique si le gestionnaire des droits d'accès aux objets (OAM) vérifie les groupes globaux lors de la détermination de l'appartenance à un groupe d'un utilisateur sous Windows.
Strophe de service	
<u>Nom</u>	Nom du service requis.
<u>EntryPoints</u>	Nombre de points d'entrée définis pour le service.
 <u>SecurityPolicy</u>	Sous Windows, règle de sécurité pour chaque gestionnaire de files d'attente
  <u>SecurityPolicy</u>	Sous AIX and Linux, indique si le gestionnaire de files d'attente utilise une autorisation basée sur l'utilisateur ou sur le groupe. A partir de IBM MQ 9.3.0, vous pouvez également créer un nom d'utilisateur qui n'est pas un nom d'utilisateur de système d'exploitation.
<u>SharedBindingsUserId</u>	Pour les liaisons partagées uniquement, que la zone UserIdentifier de la structure IdentityContext , de la fonction MQZ_AUTHENTICATE_USER, soit l'ID utilisateur effectif ou l'ID utilisateur réel.
<u>FastpathBindingsUserId</u>	Pour les liaisons de chemin d'accès rapide uniquement, que la zone UserIdentifier de la structure IdentityContext de la fonction MQZ_AUTHENTICATE_USER soit l'ID utilisateur effectif ou l'ID utilisateur réel.
<u>IsolatedBindingsUserId</u>	Pour les liaisons isolées uniquement, que la zone UserIdentifier de la structure IdentityContext de la fonction MQZ_AUTHENTICATE_USER soit l'ID utilisateur effectif ou l'ID utilisateur réel.
sectionServiceComponent	
<u>Service</u>	Nom du service requis.
<u>Nom</u>	Nom descriptif du composant de service.
<u>Module</u>	Nom du module devant contenir le code de ce composant.
<u>ComponentDataTaille</u>	Taille, en octets, de la zone de données de composant transmise au composant lors de chaque appel.
 Strophe SPX	
 <u>Socket</u>	Nombre de sockets SPX en notation hexadécimale.
 <u>BoardNum</u>	Numéro de carte de réseau local.
 <u>KeepAlive</u>	Activez ou désactivez la fonction KeepAlive .

Tableau 12. Sections du fichier *qm.ini* (suite)







Strophe et attributs	Description des attributs
 <u>Library1</u>	Nom de la DLL SPX.
 <u>Library2</u>	Identique à <u>LibraryName1</u> , utilisé si le code est stocké dans deux bibliothèques distinctes.
 <u>ListenerBacklog</u>	Remplacez le nombre par défaut de demandes en attente pour le programme d'écoute SPX.
Strophe SSL	
<u>OutboundSNI</u>	Indique si les clients compatibles SNI définiront SNI sur le nom de canal IBM MQ cible sur le système distant lors de l'établissement d'une connexion TLS ou sur le nom d'hôte.
<u>AllowOutboundSNI</u>	Indique si les clients compatibles SNI définiront SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS.  Avertissement :  Depuis la IBM MQ 9.3.0, cette propriété est obsolète. Utilisez OutboundSNI à la place.
<u>SpécificationsAllowedCipher</u>	Indique une liste personnalisée de CipherSpecs qui sont commandés et activés pour être utilisés avec les canaux IBM MQ sur Multiplatforms.
<u>AllowTLSV13</u>	Indique si un gestionnaire de files d'attente peut utiliser les CipherSpecsTLS 1.3.
<u>CDPCheckExtensions</u>	Indique si les canaux TLS de ce gestionnaire de files d'attente tentent de vérifier les serveurs CDP nommés dans les extensions de certificat de point CrDistribution.
<u>MinimumRSAKeyTaille</u>	Indique la taille de clé minimale que les certificats RSA doivent avoir pour être acceptés.
<u>Authentification OCSPAuthentication</u>	Action à effectuer lorsqu'un statut de révocation ne peut pas être déterminé à partir d'un serveur OCSP.
<u>OCSPCheckExtensions</u>	Indique si les canaux TLS de ce gestionnaire de files d'attente tentent de vérifier les serveurs OCSP nommés dans les extensions de certificat d'accès AuthorityInfo.
<u>Délai d'attente OCSPTimeout</u>	Nombre de secondes d'attente d'un répondeur OCSP lors de l'exécution d'une vérification de révocation.
 <u>PeerCertChainValidation</u>	Paramètre de validation de certificat IBM Global Security Kit (GSKit).
<u>SSLHTTPProxyName</u>	Nom d'hôte ou adresse réseau du serveur proxy HTTP qui doit être utilisé par GSKit pour les vérifications OCSP.
<u>SSLHTTPConnectTimeout</u>	Nombre de secondes d'attente pour l'établissement d'une connexion réseau à un serveur HTTP lors de l'exécution d'une vérification de révocation.
Strophe Subpool «3» , à la page 123	Cette section est créée par IBM MQ. Ne la modifiez pas.

Tableau 12. Sections du fichier *qm.ini* (suite)




Strophe et attributs	Description des attributs
<u>ShortSubpool</u> «3», à la page 123	Nom correspondant à un répertoire et à un lien symbolique créés dans le répertoire <code>/var/mqm/sockets</code> , que IBM MQ utilise pour les communications internes entre ses processus en cours d'exécution.
 Strophe TCP	
<u>Port</u>	Numéro de port par défaut, en notation décimale, pour les sessions TCP/IP.
 <u>Library1</u>	Nom de la DLL des sockets TCP/IP.
<u>KeepAlive</u>	Activez ou désactivez la fonction KeepAlive .
<u>ListenerBacklog</u>	Remplacez le nombre par défaut de demandes en attente pour le programme d'écoute TCP/IP.
<u>Connect_Timeout</u>	Nombre de secondes avant qu'une tentative de connexion du socket n'expire.
<u>TailleSndBuff</u>	Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité émettrice des canaux.
<u>RcvBuffTaille</u>	Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité réceptrice des canaux.
<u>RcvSndBuffSize</u>	Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité émettrice d'un canal récepteur.
<u>RcvRcvBuffSize</u>	Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité réceptrice d'un canal récepteur.
<u>SvrSndBuffSize</u>	Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité serveur d'un canal de connexion serveur de connexion client.
<u>SvrRcvBuffSize</u>	Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité serveur d'un canal de connexion serveur de connexion client.
 <u>SecureCommsuniquement</u>	Indique si la communication en texte en clair est autorisée, la valeur par défaut ou non autorisée.
Strophe Paramètres d'optimisation	
<u>SuppressDspAuthFail</u>	Indique si le gestionnaire de files d'attente supprime la génération d'événements d'autorisation et l'écriture de messages d'erreur AMQ8077 dans le journal des erreurs en cas d'échec d'une vérification d'autorisation, si la connexion ne dispose pas des droits + dsp sur un objet.
<u>ImplSyncOpenOutput</u>	Nombre minimal d'applications pour lesquelles la file d'attente est ouverte pour insertion, avant qu'un point de synchronisation implicite puisse être activé pour une insertion persistante, en dehors du point de synchronisation.
<u>UniformClusterNom</u>	Nom du cluster IBM MQ que vous utilisez en tant que cluster uniforme.

Tableau 12. Sections du fichier *qm.ini* (suite)

Strophe et attributs	Description des attributs
<u>OAMLdapConnectDélai d'attente</u>	Durée maximale, en secondes, pendant laquelle le client LDAP attend pour établir une connexion TCP au serveur.
<u>OAMLdapQueryTimeLimit</u>	Durée maximale, en secondes, pendant laquelle le client LDAP attend une réponse à une demande LDAP du serveur.
<u>OAMLdapResponseWarningTime</u>	Si une connexion à un serveur LDAP a pris plus de temps que le nombre de secondes défini par le paramètre OAMLdapResponseWarningTime , un message <u>AMQ5544W</u> est consigné dans le journal des erreurs.
<u>ExpiryInterval</u>	Indique la fréquence à laquelle le gestionnaire de files d'attente analyse les files d'attente à la recherche de messages arrivés à expiration qui n'ont pas encore été nettoyés par d'autres activités de file d'attente. Il s'agit d'un intervalle de temps en secondes.
<u>LivenessHeartBeatLen</u>	Configure la fréquence à laquelle les vérifications du gestionnaire de files d'attente qui écrivent dans le journal sont effectuées à une fréquence raisonnable.
<u>ECHearBeatLen</u>	Configure la fréquence des diagnostics d'intégrité généraux du gestionnaire de files d'attente.
<u>FileLockHeartBeatLen</u>	Modifie la valeur par défaut du verrou de fichier pour un gestionnaire de files d'attente multi-instance que le contrôleur d'exécution exécute régulièrement afin de s'assurer qu'il détient toujours le verrou exclusif sur le fichier multi-instance principal.
Strophe Variables	
<u>attribut=valeur</u>	Nom et valeur associée à utiliser comme insertion lors des définitions MQSC.
StropheXAResourceManager	
<u>Nom</u>	Instance du gestionnaire de ressources.
<u>SwitchFile</u>	Nom qualifié complet du fichier de chargement contenant la structure de commutateur XA du gestionnaire de ressources.
<u>XAOpenString</u>	Chaîne de données à transmettre au point d'entrée <u>xa_open</u> du gestionnaire de ressources.
<u>XACloseString</u>	Chaîne de données à transmettre au point d'entrée <u>xa_close</u> du gestionnaire de ressources.
<u>Unité d'exécution de contrôle</u>	Valeur utilisée par le gestionnaire de files d'attente pour la sérialisation lorsqu'il doit appeler le gestionnaire de ressources à partir de l'un de ses propres processus à unités d'exécution multiples. Obligatoire pour Windows.

Remarques :

1. La section AccessMode est définie par l'option **-a [r]** dans la commande **crtmqm**. Ne modifiez pas la section AccessMode après la création du gestionnaire de files d'attente.

2. La section RestrictedMode est définie par l'option **-g** dans la commande **crtmqm** . Ne modifiez pas cette section après la création du gestionnaire de files d'attente. Si vous n'utilisez pas l'option **-g** , la section n'est pas créée dans le fichier `qm.ini` .
3. La strophe Subpool et l'attribut ShortSubpoolName dans cette strophe sont écrits automatiquement par IBM MQ lorsque vous créez un gestionnaire de files d'attente. IBM MQ choisit une valeur pour le nom ShortSubpool. Ne modifiez pas cette valeur.

Windows Section AccessMode du fichier `qm.ini`

Le mode d'accès s'applique uniquement aux serveurs Windows . La section AccessMode du fichier `qm.ini` est définie par l'option `-a [r]` de la commande **crtmqm** . Ne modifiez pas la section AccessMode après la création du gestionnaire de files d'attente.

Utiliser le groupe d'accès (`-a [r]`) de la commande **crtmqm** pour spécifier un groupe de sécurité Windows , dont les membres auront un accès complet à tous les fichiers de données du gestionnaire de files d'attente. Le groupe peut être un groupe local ou global, selon la syntaxe utilisée. La syntaxe valide pour le nom de groupe est la suivante:

LocalGroup
Nom de domaine\GlobalGroup nom
GlobalGroup nom @ Nom de domaine

Vous devez définir le groupe d'accès supplémentaire avant d'exécuter la commande **crtmqm** avec l'option `-a [r]` .

Si vous spécifiez le groupe à l'aide de `-ar` au lieu de `-a`, le groupe `mqm` local n'est pas autorisé à accéder aux fichiers de données du gestionnaire de files d'attente. Utilisez cette option si le système de fichiers hébergeant les fichiers de données du gestionnaire de files d'attente ne prend pas en charge les entrées de contrôle d'accès pour les groupes définis en local.

Le groupe est généralement un groupe de sécurité global utilisé pour fournir des gestionnaires de files d'attente multi-instances ayant accès à un gestionnaire de files d'attente partagé et au dossier des journaux. Utilisez le groupe d'accès de sécurité supplémentaire pour définir des droits de lecture et d'écriture sur le dossier ou partager le gestionnaire de files d'attente contenant les données et les fichiers journaux.

Le groupe d'accès de sécurité supplémentaire est une alternative à l'utilisation du groupe local nommé `mqm` pour définir les droits sur le dossier contenant les données et les journaux du gestionnaire de files d'attente. Contrairement au groupe local `mqm`, vous pouvez faire du groupe d'accès de sécurité supplémentaire un groupe local ou global. Il doit s'agir d'un groupe global pour définir des droits sur les dossiers partagés qui contiennent les données et les fichiers journaux utilisés par des gestionnaires de files d'attente multi-instances.

Le système d'exploitation Windows vérifie les droits d'accès permettant de lire et d'écrire des données et des fichiers journaux du gestionnaire de files d'attente. Il vérifie les droits de l'ID utilisateur qui exécute les processus de gestionnaire de files d'attente. L'ID utilisateur qui est vérifié varie selon que vous avez démarré le gestionnaire de files d'attente en tant que service ou en mode interactif. Si vous avez démarré le gestionnaire de files d'attente en tant que service, l'ID utilisateur vérifié par le système d'exploitation Windows est celui que vous avez configuré avec l'assistant de **préparation** IBM MQ. Si vous avez démarré le gestionnaire de files d'attente en mode interactif, l'ID utilisateur vérifié par le système d'exploitation Windows est celui qui a exécuté la commande **strmqm**.

L'ID utilisateur doit être membre du groupe `mqm` local pour démarrer le gestionnaire de files d'attente. Si l'ID utilisateur est membre du groupe d'accès de sécurité supplémentaire, le gestionnaire de files d'attente peut lire et écrire des fichiers auxquels sont attribués des droits à l'aide du groupe.

Restriction : Vous pouvez spécifier un groupe d'accès de sécurité supplémentaire sur le système d'exploitation Windows seulement. Si vous spécifiez un groupe d'accès de sécurité supplémentaire sur d'autres systèmes d'exploitation, la commande **crtmqm** renvoie une erreur.

Exemple de section

```
AccessMode:  
SecurityGroup=wmq\wmq
```

Concepts associés

«Fichiers et répertoires de données et de journaux du gestionnaire de files d'attente non partagés sécurisés sous Windows», à la page 583

«Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows», à la page 579

Tâches associées

«Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows», à la page 553

Référence associée

[crtmqm \(création du gestionnaire de files d'attente\)](#)

Multi

ApiExitStrophe locale du fichier qm.ini

La section locale ApiExitspécifie les routines d'exit d'API pour un gestionnaire de files d'attente.

Pour un serveur, modifiez la section locale ApiExitdu fichier `qm.ini` afin d'identifier les routines d'exit d'API pour un gestionnaire de files d'attente.

Windows

Linux

Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page des propriétés du gestionnaire de files d'attente IBM MQ Explorer Exits .

Pour un client, modifiez la strophe locale ApiExitdans le fichier `mqclient.ini` afin d'identifier les routines d'exit d'API pour un gestionnaire de files d'attente.

Présentation

La section `ApiExitLocal` ne permet de spécifier qu'un seul `Module` , mais quatre modules doivent être fournis, comme suit:

- 32 bits sans unités d'exécution
- Unités d'exécution 32 bits
- 64 bits sans unités d'exécution
- Unités d'exécution 64 bits

Notez que IBM MQ ajoute `_r` au nom de module fourni pour identifier la version à unités d'exécution de l'exit, mais IBM MQ ne fournit pas de mécanisme directement équivalent pour les variantes 32 bits et 64 bits.

Les versions de `amqsaxe0` et de `amqsaxe0_r` fournies dans `prefix/mqm/samp/bin` sont générées pour la taille native du gestionnaire de files d'attente sur la plateforme pour laquelle elles sont générées (à présent, toutes les versions 64 bits) et ne peuvent être utilisées que par les applications s'exécutant dans la même taille native.

Si un nom de module non qualifié est fourni, IBM MQ recherche dans `/var/mqm/exits` les variantes 32 bits et dans `/var/mqm/exits64` les variantes 64 bits.

Par exemple, `module=amqsaxe` implique:

```
/var/mqm/exits/amqsaxe - 32 bit unthreaded variant  
/var/mqm/exits/amqsaxe_r - 32 bit threaded variant  
/var/mqm/exits64/amqsaxe - 64 bit unthreaded variant  
/var/mqm/exits64/amqsaxe_r - 64 bit threaded variant
```

Windows

Sur les systèmes Windows, vous pouvez également utiliser la commande **amqmdain** pour modifier les entrées des exits API. (Pour identifier les routines d'exit d'API pour tous les gestionnaires de files d'attente, utilisez les sections `ApiExitCommon` et `ApiExitTemplate`, comme décrit dans «[ApiExitSections de modèle communes et ApiExitdu fichier mqs.ini](#)», à la page 104.)

Notez que pour que l'exit API fonctionne correctement, le message du serveur doit être envoyé au client non converti. Une fois que l'exit API a traité le message, le message doit être converti sur le client. Par conséquent, vous devez avoir installé tous les exits de conversion sur le client.

Pour plus d'informations sur l'utilisation de ces attributs, voir [Configuration des exits API](#).

Paramètres

Name=nom_exit_pix

Nom descriptif de l'exit API qui lui a été transmis dans la zone `ExitInfoNom` de la structure `MQAXP`.

Ce nom doit être unique, ne pas dépasser 48 caractères et contenir uniquement des caractères valides pour les noms d'objets IBM MQ (par exemple, les noms de file d'attente).

Function=nom_fonction

Nom du point d'entrée de fonction dans le module contenant le code d'exit d'API. Ce point d'entrée est la fonction `MQ_INIT_EXIT`.

La longueur de cette zone est limitée à `MQ_EXIT_NAME_LENGTH`.

Module=nom_module

Module contenant le code d'exit d'API.

Ce champ est utilisé tel quel s'il contient le chemin d'accès complet au module. Si cette zone contient uniquement le nom du module, le module est localisé à l'aide de l'attribut **ExitsDefaultPath** dans la strophe `ExitPath` du fichier `qm.ini`.

Sur les plateformes qui prennent en charge des bibliothèques à unités d'exécution distinctes, vous devez fournir à la fois une version non à unités d'exécution et une version à unités d'exécution du module d'exit API. La version à unités d'exécution doit avoir un suffixe `_r`. La version à unités d'exécution du module de remplacement d'application IBM MQ ajoute implicitement `_r` au nom de module donné avant son chargement.

La longueur de cette zone est limitée à la longueur de chemin maximale prise en charge par la plateforme.

Data=nom_données

Données à transmettre à l'exit API dans la zone `ExitData` de la structure `MQAXP`.

Si vous incluez cet attribut, les blancs de début et de fin sont supprimés, la chaîne restante est tronquée à 32 caractères et le résultat est transmis à l'exit. Si vous omettez cet attribut, la valeur par défaut de 32 blancs est transmise à l'exit.

La longueur maximale de cette zone est de 32 caractères.

Sequence=numéro_séquence

Séquence dans laquelle cet exit d'API est appelé par rapport à d'autres exits d'API. Un exit avec un numéro de séquence faible est appelé avant un exit avec un numéro de séquence plus élevé. Il n'est pas nécessaire que la numérotation de séquence des exits soit contiguë. Une séquence de 1, 2, 3 a le même résultat qu'une séquence de 7, 42, 1096. Si deux exits ont le même numéro de séquence, le gestionnaire de files d'attente décide lequel appeler en premier. Vous pouvez déterminer qui a été appelé après l'événement en plaçant l'heure ou un marqueur dans la zone `ExitChain` indiquée par `ExitChainAreaPtr` dans `MQAXP` ou en écrivant votre propre fichier journal.

Cet attribut est une valeur numérique non signée.

Exemple de section

```
ApiExitLocal:  
  Name=ClientApplicationAPIchecker  
  Sequence=3
```

La section **AuthToken** permet de configurer le gestionnaire de files d'attente pour qu'il valide les jetons d'authentification fournis par les applications de connexion. Si votre service d'authentification prend en charge un noeud final JWKS pour la configuration des clés, il s'agit généralement d'une option préférable.

Pour plus d'informations, voir [Configuration d'un gestionnaire de files d'attente pour l'acceptation de jetons d'authentification à l'aide d'un noeud final JWKS](#).

La section AuthToken

KeyStore= chaîne

Chemin d'accès au magasin de clés qui contient les certificats de clé publique et les clés symétriques de l'émetteur de confiance. Vous pouvez ajouter les clés à un magasin de clés existant ou en créer un nouveau. Pour plus d'informations, voir [Configuration d'un gestionnaire de files d'attente pour l'acceptation de jetons d'authentification](#). Le gestionnaire de files d'attente utilise les clés du magasin de clés pour vérifier que le jeton d'authentification présenté par l'application est signé par l'émetteur sécurisé.

Vous pouvez utiliser un magasin de clés CMS avec l'extension de fichier .kdb ou un magasin de clés PKCS#12 avec l'extension de fichier .p12. Si le fichier de clés n'existe pas ou qu'il est inaccessible, une erreur AMQ7076E: Valeur non valide pour l'attribut dans le fichier ini est générée dans le journal des erreurs du gestionnaire de files d'attente.

Vérifiez que le type de magasin de clés correspond à l'extension de nom de fichier du magasin de clés. IBM MQ détecte le format correct du magasin de clés ; des incohérences peuvent cependant entraîner d'autres problèmes d'administration, si le type de magasin de clés et l'extension de nom de fichier ne correspondent pas.

La longueur maximale du chemin d'accès au fichier de clés est de 256 caractères.

KeyStorePwdFile= chaîne

Chemin d'accès au fichier qui contient le mot de passe chiffré pour le magasin de clés. Le fichier doit contenir le mot de passe chiffré sur une seule ligne de texte. Les mots de passe en texte en clair ne sont pas acceptés.

Utilisez la commande **runqmcured** pour chiffrer le mot de passe avant de l'enregistrer dans le fichier de mots de passe du magasin de clés. Le fichier de mots de passe du magasin de clés doit contenir uniquement le mot de passe chiffré créé à l'aide de la commande **runqmcured**.

La longueur maximale du mot de passe en texte en clair avant d'être chiffré est de 1024 caractères.

Ce paramètre est facultatif. S'il n'est pas fourni, le gestionnaire de files d'attente recherche un fichier de dissimulation avec le mot de passe dans le même répertoire et avec le même nom que le magasin de clés, mais avec l'extension de fichier .sth. Si le fichier de dissimulation est introuvable, la configuration est rejetée et le message d'erreur AMQ7006E est consigné dans le journal des erreurs du gestionnaire de files d'attente. Pour plus d'informations sur les options de stockage des mots de passe de magasin de clés, voir [Chiffrement des mots de passe de référentiel de clés](#).

La longueur maximale du chemin d'accès au fichier de mots de passe est de 256 caractères.

CertLabel= chaîne

Libellé de certificat d'un certificat de clé publique ou d'une clé symétrique dans le magasin de clés utilisé pour valider les jetons d'authentification. Vous pouvez fournir jusqu'à 32 libellés de certificat en répétant l'attribut **CertLabel**.

Lorsque vous ajoutez des certificats au magasin de clés du gestionnaire de files d'attente, attribuez-leur des libellés significatifs. Les libellés de certificat sont sensibles à la casse. Ils peuvent contenir des caractères alphanumériques, des caractères de ponctuation et des espaces. Si un caractère non valide est détecté, une erreur est renvoyée et un message d'erreur est consigné dans le journal des erreurs IBM MQ.

Les émetteurs de jetons de confiance peuvent fournir plusieurs certificats de clé publique et des clés symétriques. Par exemple, les certificats de clé publique ont des périodes de validité. Lorsqu'ils sont sur le point d'expirer, l'émetteur du jeton fournit un nouveau certificat avec une nouvelle date d'expiration. Pendant un certain temps, les deux certificats peuvent être valides.

Lorsque des applications présentent des jetons pour l'authentification, la liste des **CertLabels** est vérifiée jusqu'à ce qu'une clé valide utilisée pour signer le jeton soit trouvée. Si la correspondance est trouvée, la signature du jeton est validée.

Si **CertLabel** n'est pas spécifié, la connexion de l'application qui présente le jeton échoue avec le code anomalie 2063 MQRC_SECURITY_ERROR et le message AMQ5786E: Authentication token configuration error est consigné dans le journal des erreurs du gestionnaire de files d'attente.

La longueur maximale du libellé de certificat est de 64 caractères.

Exemple :

```
AuthToken:  
  KeyStore=/var/mqm/qmgrs/qmgrs/qm1/tokenissuer/key.kdb  
  KeyStorePwdFile=/var/mqm/qmgrs/qmgrs/qm1/tokenissuer/key.pw  
  CertLabel=token  
  CertLabel=rsakey  
  CertLabel=mark  
  ... up to 32 CertLabel fields
```

UserClaim= chaîne

Réclamation dans le jeton qui contient l'ID utilisateur adopté par le gestionnaire de files d'attente pour les vérifications d'autorisation.

Ce paramètre est facultatif si le gestionnaire de files d'attente est configuré avec **ADOPTCTX(NO)**. Si **ADOPTCTX(YES)** est utilisé, ce paramètre est obligatoire. **ADOPTCTX** est un attribut présent dans l'objet d'informations d'authentification (AUTHINFO) référencé par l'attribut **CONNAUTH** du gestionnaire de files d'attente.

Pour adopter une identité, le jeton doit contenir une revendication dont le nom est spécifié dans l'attribut **UserClaim** de la section **AuthToken** et **ADOPTCTX(YES)** doit être utilisé.

Par exemple, si votre jeton inclut une revendication "AppUser": "MyUserName", vous devez spécifier **UserClaim=AppUser** dans la section **AuthToken** du fichier `qm.ini` afin d'adopter l'identité "MyUserName" pour l'autorisation.

La longueur maximale de la valeur d'attribut **UserClaim** est de 128 caractères.

Remarque : Depuis la IBM MQ 9.4.0, si la section **AuthToken** est spécifiée, la valeur effective de l'attribut **SecurityPolicy** de la section **Service** est définie sur **UserExternal**. L'authentification par jeton n'est pas disponible si **SecurityPolicy** est explicitement défini sur **Group** dans la section **Service**. Si **SecurityPolicy** est défini sur **Groupe**, supprimez l'attribut **SecurityPolicy** de la strophe **Service**, puis redémarrez le gestionnaire de files d'attente. Pour plus d'informations, voir [SecurityPolicy](#).

Remarque : Utilisez l'attribut **ADOPTCTX** de l'objet d'informations d'authentification pour contrôler si l'ID utilisateur dans le jeton est adopté pour les vérifications d'autorisation. Lorsque vous créez le gestionnaire de files d'attente, cet attribut est défini sur **ADOPTCTX(YES)**. Cette valeur entraîne l'adoption de l'ID utilisateur du jeton. L'ID utilisateur doit répondre aux exigences relatives aux ID utilisateur dans les jetons d'authentification. Pour plus d'informations, voir [ID utilisateur dans les jetons d'authentification](#). Si la demande d'utilisateur de jeton contient un ID utilisateur qui ne répond pas aux exigences, la connexion est rejetée avec le code anomalie 2035 MQRC_NOT_AUTHORIZED. Si **ADOPTCTX(NO)** est défini, le jeton est utilisé uniquement pour l'authentification et un autre utilisateur doit être utilisé pour l'autorisation.

AllowOSGroups=NO (valeur par défaut) |YES

La valeur par défaut est NO. Détermine si une identité adoptée à partir d'un jeton est traitée comme un utilisateur du système d'exploitation (OS) et si les appartenances au groupe de l'utilisateur du système d'exploitation correspondant sont honorées lors de l'autorisation.

AllowOSGroups= NON | N

Les vérifications d'autorisation sont basées uniquement sur le nom de l'utilisateur adopté à partir du jeton.

AllowOSGroups= YES | Y

Les vérifications d'autorisation sont basées sur le nom de l'utilisateur et les groupes auxquels ils peuvent appartenir sont également vérifiés.

Remarque : L'attribut **allowOSGroups** de la section **AuthToken** prend toujours effet tout au long de l'authentification par jeton, même si toutes les autres configurations de validation de jeton sont gérées via la section **JWKS**.

Exemple de section-authentification uniquement

Cette section peut contenir uniquement l'attribut **AllowOSGroups**. Toutefois, si la configuration du magasin de clés local est incluse, elle doit contenir au minimum:

- Chemin d'accès au fichier **KeyStore** et
- Nom **CertLabel**.

```
AuthToken:  
  KeyStore=/var/mqm/qmgrs/qmgrs/qm1/tokenissuer/key.kdb  
  CertLabel=token  
  ... up to 32 CertLabel fields
```

Si vous n'avez inclus que les deux paramètres minimum, procédez comme suit:

- Un fichier de dissimulation `key.sth` doit exister avec le mot de passe de magasin de clés chiffré pour que le fichier de mot de passe de magasin de clés ne soit pas requis.
- Le jeton ne contient pas de nom d'utilisateur qui doit être transmis à IBM MQ pour autorisation. L'application peut se connecter et être authentifiée, mais un mécanisme différent doit être en place pour permettre à l'application d'effectuer un travail après sa connexion.

Selon la configuration de votre gestionnaire de files d'attente, le nom d'utilisateur utilisé pour l'autorisation peut être celui défini sur le canal via les règles MCA ou le nom d'utilisateur que l'application client a exécuté sur votre serveur et qui peut appartenir à des groupes disposant de droits d'accès. Gardez à l'esprit que lorsque vous utilisez des jetons:

- Votre gestionnaire de files d'attente est placé en mode **UserExternal**, ce qui signifie que les utilisateurs qui n'existent pas sur le système d'exploitation sur lequel le gestionnaire de files d'attente est en cours d'exécution peuvent être utilisés pour l'authentification.
- Même si vous n'incluez pas l'option **AllowOSGroups** dans votre section **AuthToken** `qm.ini`, la valeur par défaut est No. Par conséquent, si vous incluez **UserClaim** mais que vous ne spécifiez pas **AllowOSGroups=Yes**, l'utilisateur de jeton adopté pour l'autorisation n'est pas vérifié pour les groupes auxquels il peut appartenir sur le système d'exploitation sur lequel le gestionnaire de files d'attente s'exécute.

Exemple de section-authentification et autorisation

Vous pouvez définir tous les paramètres **AuthToken** :

- Chemin d'accès au fichier **KeyStore**,
- Chemin d'accès au fichier **KeyStorePwdFile**,
- Nom **CertLabel**,
- Nom **UserClaim** et
- l'option **AllowOSGroups**.

```
AuthToken:  
  KeyStore=/var/mqm/qmgrs/qmgrs/QMJWT/ssl/key.kdb  
  KeyStorePwdFile=/var/mqm/qmgrs/QMJWT/ssl/key.pw  
  CertLabel=token
```



```
CertLabel=rsaKey
CertLabel=mark
... up to 32 CertLabel fields
UserClaim=AppUser
AllowOSGroups=Y
```

Si vous avez inclus tous les paramètres disponibles, procédez comme suit:

- Chiffrez le mot de passe du magasin de clés à l'aide de la commande **runqmcrcd** . Sauvegardez-le dans un fichier, puis incluez le chemin d'accès au fichier dans la section **AuthToken** .
- Le nom d'utilisateur figurant dans la revendication de l'utilisateur du jeton d'authentification est utilisé à la fois pour l'authentification et l'autorisation.
 - L'utilisateur de jeton peut exister en tant qu'utilisateur sur le système d'exploitation où s'exécute le gestionnaire de files d'attente.
 - Vous avez défini un objet d'informations d'authentification pour activer la vérification des utilisateurs.
 - Vous configurez des enregistrements d'authentification de canal pour adopter un utilisateur autorisé à interagir avec des objets IBM MQ , en fonction de l'authentification de canal ou des règles MCA.

Votre stratégie d'authentification et d'autorisation des utilisateurs de jeton dépend de vos exigences et de la manière dont vos gestionnaires de files d'attente IBM MQ sont déjà configurés. Pour plus d'informations, voir [Utilisation de jetons d'authentification](#).

Concepts associés

[Utilisation des jetons](#)

Tâches associées

[Configuration d'un gestionnaire de files d'attente pour qu'il accepte **AuthTokens**](#)

[Utilisation de jetons d'authentification dans une application](#)

Multi

Section AutoCluster du fichier qm.ini

La section AutoCluster est utilisée lorsque le gestionnaire de files d'attente commence à identifier si le cluster est membre d'un cluster automatique et peut identifier les référentiels complets du cluster.

Les attributs suivants sont obligatoires pour la section AutoCluster :

Type = *Uniforme*

Indique le type de cluster automatique et la seule option valide est *Uniforme*, qui représente un cluster uniforme.

ClusterName=< *Chaîne*>

Nom du cluster, c'est-à-dire le nom de cluster automatique.

Les attributs suivants sont facultatifs pour la section AutoCluster , mais vous devez les fournir par paires:

RepositoryName1 =< *String*>

Il s'agit du nom du gestionnaire de files d'attente pour le premier référentiel complet du cluster automatique. Il peut s'agir du nom de ce gestionnaire de files d'attente ou d'un autre nom.

Repository1Conname=< *chaîne de nom de connexion* >

Il s'agit de la valeur du nom de connexion (CONNNAME) pour la façon dont les membres du cluster automatique doivent se connecter à ce gestionnaire de files d'attente.

Repository2Name=< *String*>

Il s'agit du nom du gestionnaire de files d'attente pour le deuxième référentiel complet dans le cluster automatique. Il peut s'agir du nom de ce gestionnaire de files d'attente ou d'un autre nom.

Repository2Conname=< *chaîne de nom de connexion* >

Il s'agit de la valeur du nom de connexion (CONNNAME) pour la façon dont les membres du cluster automatique doivent se connecter à ce gestionnaire de files d'attente.

Exemple de section

```
AutoCluster:  
  Repository1Name=QM1  
  Repository2Name=QM2  
  Repository1Conname=127.0.0.1(1414)  
  Repository2Conname=127.0.0.1(1415)  
  ClusterName=UNIFORMCLUSTER1  
  Type=Uniform
```

Concepts associés

[«Équilibrage automatique des applications», à la page 447](#)

L'équilibrage automatique des applications améliore considérablement la distribution et la disponibilité des applications en permettant à un cluster uniforme IBM MQ de gérer étroitement la distribution des applications dans le cluster, plutôt que de s'appuyer sur la randomisation ou sur une épingle manuel des applications à des gestionnaires de files d'attente spécifiques.

Tâches associées

[«Création d'un nouveau cluster uniforme», à la page 461](#)

Comment créer un nouveau cluster uniforme.

Référence associée

[«Utilisation de la configuration de cluster automatique», à la page 465](#)

Vous configurez IBM MQ pour activer la configuration automatique en modifiant les informations de configuration `qm.ini`.

Multi

Section AutoConfig du fichier `qm.ini`

Les attributs de la section AutoConfig sont fréquemment utilisés dans le cadre de la configuration de clusters uniformes.

Remarque : Vous ne pouvez utiliser la section AutoCluster que pour les clusters uniformes.

MQSCConfig=< chemin >

Il s'agit d'un chemin d'accès complet au fichier ou d'un chemin d'accès à un répertoire dans lequel tous les fichiers `*.mqsc` sont appliqués au gestionnaire de files d'attente à chaque démarrage du gestionnaire de files d'attente.

Pour plus d'informations, voir [Configuration automatique à partir d'un script MQSC au démarrage](#).

IniConfig=< chemin >

Il s'agit d'un chemin d'accès complet ou d'un chemin d'accès à un répertoire dans lequel tous les fichiers `*.ini` sont appliqués au fichier `qm.ini` à chaque démarrage du gestionnaire de files d'attente.

Pour plus d'informations, voir [«Configuration automatique de qm.ini au démarrage», à la page 112](#).

ConfigTimeout

Valeur (en secondes) pendant laquelle le gestionnaire de files d'attente attend la fin de la configuration automatique. Après ce délai, le gestionnaire de files d'attente continue à démarrer et est disponible pour que les applications se connectent.

Le comportement par défaut est l'absence de délai d'attente. Cela signifie que le gestionnaire de files d'attente n'est pas disponible pour la connexion des applications tant que toutes les commandes de configuration automatique n'ont pas été exécutées.

Vous ne devez pas configurer cet attribut simplement parce que la configuration prend beaucoup de temps, car les applications peuvent être en mesure de se connecter avant que la configuration qui leur est appliquée ne soit terminée, par exemple, la création de files d'attente requises par l'application.

Exemple de section

```
AutoConfig:
MQSCConfig=/tmp/auto.mqsc
IniConfig=/tmp/auto.ini
ConfigTimeout=120
```

Concepts associés

«Équilibrage automatique des applications», à la page 447

L'équilibrage automatique des applications améliore considérablement la distribution et la disponibilité des applications en permettant à un cluster uniforme IBM MQ de gérer étroitement la distribution des applications dans le cluster, plutôt que de s'appuyer sur la randomisation ou sur un épinglage manuel des applications à des gestionnaires de files d'attente spécifiques.

Tâches associées

«Création d'un nouveau cluster uniforme», à la page 461

Comment créer un nouveau cluster uniforme.

Référence associée

«Utilisation de la configuration de cluster automatique», à la page 465

Vous configurez IBM MQ pour activer la configuration automatique en modifiant les informations de configuration `qm.ini`.

Multi Strophe channels du fichier `qm.ini`

Les attributs de la section Canaux déterminent la configuration d'un canal.

z/OS Ces informations ne s'appliquent pas à IBM MQ for z/OS.

Utilisez la section CHANNELS du fichier `qm.ini` pour spécifier des informations sur les canaux.

Windows **Linux** Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page des propriétés du gestionnaire de files d'attente IBM MQ Explorer Channels.

MaxChannels = 100 (valeur par défaut) | nombre

Nombre maximal de canaux *en cours* autorisés.

La valeur par défaut est 100.

Vous pouvez définir **MaxChannels** sur une valeur différente pour limiter le nombre maximal de canaux en cours si nécessaire. Pour IBM MQ Appliance, la valeur par défaut est 999 999 999 et ne doit pas être modifiée.

MaxActiveChannels = valeur_canaux_max

Nombre maximal de canaux pouvant être *actifs* à tout moment. La valeur par défaut est celle spécifiée pour l'attribut **MaxChannels**.

MaxInitiators = 3 (valeur par défaut) | nombre

Nombre maximal d'initiateurs. La valeur par défaut et maximale est 3.

MQIBindType= FASTPATH | STANDARD

La liaison pour les applications:

Fastpath

Les canaux se connectent à l'aide de MQCONNX FASTPATH ; il n'y a pas de processus d'agent.

STANDARD

Les canaux se connectent à l'aide de STANDARD.

PipeLineLength = 1 | nombre

Nombre maximal d'unités d'exécution simultanées qu'un canal utilisera. La valeur par défaut est 1. Toute valeur supérieure à 1 est traitée comme 2.

Lorsque vous utilisez le pipeline, configurez les gestionnaires de files d'attente aux deux extrémités du canal pour qu'ils aient un **PipeLineLength** supérieur à 1.

Remarque : Le pipeline n'est efficace que pour les canaux TCP/IP.

Pour plus d'informations, voir [Prise en charge de plusieurs unités d'exécution-Pipetage](#) .

AdoptNewMCA = NO (par défaut) | SVR | SDR | RCVR | CLUSRCVR | ALL | FASTPATH

Si IBM MQ reçoit une demande de démarrage d'un canal, mais détecte qu'une instance du canal est déjà en cours d'exécution, dans certains cas, l'instance de canal existante doit être arrêtée pour que la nouvelle puisse démarrer. L'attribut **AdoptNewMCA** vous permet de contrôler les types de canaux qui peuvent être arrêtés de cette manière.

Si vous spécifiez l'attribut **AdoptNewMCA** pour un type de canal particulier, mais que le nouveau canal ne démarre pas car une instance de canal correspondante est déjà en cours d'exécution:

1. Le nouveau canal tente d'arrêter le précédent en lui demandant de s'arrêter.
2. Si le serveur de canal précédent ne répond pas à cette demande à l'expiration de l'intervalle d'attente **AdoptNewMCATimeout** , l'unité d'exécution ou le processus du serveur de canal précédent est arrêté.
3. Si le serveur de canaux précédent ne s'est pas arrêté après l'étape 2 et que le délai d'attente **AdoptNewMCATimeout** expire une seconde fois, IBM MQ arrête le canal avec une erreur CHANNEL IN USE .

La fonctionnalité **AdoptNewMCA** s'applique aux canaux serveur, émetteur, récepteur et récepteur de cluster. Dans le cas d'un canal émetteur ou serveur, une seule instance d'un canal portant un nom particulier peut être exécutée dans le gestionnaire de files d'attente de réception. Dans le cas d'un canal récepteur ou récepteur de cluster, plusieurs instances d'un canal portant un nom particulier peuvent être exécutées dans le gestionnaire de files d'attente récepteur, mais une seule instance peut être exécutée à la fois à partir d'un gestionnaire de files d'attente distant particulier.

Remarque : **AdoptNewMCA** n'est pas pris en charge sur les canaux de connexion serveur ou demandeur.

Indiquez une ou plusieurs valeurs, séparées par des virgules ou des blancs, dans la liste suivante:

NO

La fonction **AdoptNewMCA** n'est pas requise. Il s'agit de l'option par défaut.

SVR

Adopter les canaux serveur.

SDR

Adopter les canaux émetteurs.

RCVR

Adopter les canaux récepteurs.

CLUSRCVR

Adopter les canaux récepteurs de cluster.

TOUT

Adopter tous les types de canal à l'exception des canaux FASTPATH.

Fastpath

Adoptez le canal s'il s'agit d'un canal FASTPATH. Cela se produit uniquement si le type de canal approprié est également spécifié, par exemple: **AdoptNewMCA=RCVR, SVR, FASTPATH**.

Attention ! : L'attribut **MCA AdoptNew** peut se comporter de manière imprévisible avec les canaux FASTPATH. Soyez très prudent lorsque vous activez l'attribut **MCA AdoptNew** pour les canaux FASTPATH.

AdoptNewMCATimeout= 60 (par défaut) | 1-3600

Durée, en secondes, pendant laquelle la nouvelle instance de canal attend l'arrêt de l'ancienne instance de canal. Indiquez une valeur comprise entre 1 et 3600. La valeur par défaut est 60.

AdoptNewMCACheck = QM | ADDRESS | NAME | ALL

Type de vérification requis lors de l'activation de l'attribut **AdoptNewMCA** . Si possible, effectuez une vérification complète afin de protéger vos canaux de l'arrêt, par inadvertance ou de manière malveillante. Vérifiez au moins que les noms de canal correspondent.

Spécifiez une ou plusieurs des valeurs suivantes, séparées par des virgules ou des blancs dans le cas de *QM*, *NAME* ou *ALL*:

QM

Vérifiez que les noms de gestionnaire de files d'attente correspondent.

Notez que le nom du gestionnaire de files d'attente lui-même est mis en correspondance, et non le QMID.

ADDRESS

Vérifiez l'adresse IP de la source de communication. Par exemple, l'adresse TCP/IP.

Remarque : Les valeurs CONNAME séparées par des virgules s'appliquent aux adresses cible et ne sont donc pas pertinentes pour cette option.

Dans le cas où un gestionnaire de files d'attente multi-instance bascule de *hosta* vers *hostb*, tous les canaux sortants de ce gestionnaire de files d'attente utiliseront l'adresse IP source d'*hostb*. S'il est différent de *hosta*, `AdoptNewMCACheck=ADDRESS` ne correspond pas.

Vous pouvez utiliser SSL ou TLS avec authentification mutuelle pour empêcher un agresseur d'interrompre un canal existant en cours d'exécution. Vous pouvez également utiliser une solution de type HACMP avec une reprise IP au lieu de gestionnaires de files d'attente multi-instance ou utiliser un équilibreur de charge réseau pour masquer l'adresse IP source.

NOM

Vérifiez que les noms de canal correspondent.

TOUT

Recherchez les noms de gestionnaire de files d'attente correspondants, l'adresse de communication et les noms de canal correspondants.

La valeur par défaut est `AdoptNewMCACheck=NAME, ADDRESS, QM`.

ChlauthEarlyAdopt = Y (par défaut) | N

L'ordre dans lequel les règles d'authentification de connexion et d'authentification de canal sont traitées est un facteur important dans la détermination du contexte de sécurité pour les connexions d'application client IBM MQ .



Avertissement : La valeur par défaut si **ChlauthEarlyAdopt** n'est pas présent dans le fichier `qm.ini` est N. Toutefois, tous les gestionnaires de files d'attente sont créés avec **ChlauthEarlyAdopt=Y** automatiquement ajouté au fichier `qm.ini` .

ChlauthEarlyAdopt n'adopte que les ID utilisateur qui ont été fournis à un gestionnaire de files d'attente pour l'authentification de connexion, si `ADOPTCTX (YES)` est défini sur l'objet `AUTHINFO` d'authentification de connexion sur le gestionnaire de files d'attente.

Les valeurs admises pour **ChlauthEarlyAdopt** sont les suivantes:

Y

Le canal valide et adopte les données d'identification d'ID utilisateur et de mot de passe qui ont été fournies par une application à l'aide de l'authentification de connexion de gestionnaire de files d'attente avant d'appliquer les règles d'authentification de canal. Dans ce mode de fonctionnement, les règles d'authentification de canal correspondent à l'ID utilisateur résultant des vérifications d'authentification de connexion.

N

Le canal retarde la validation de l'authentification de connexion des données d'identification d'ID utilisateur et de mot de passe qui ont été fournies par une application jusqu'à ce que les règles d'authentification de canal aient été appliquées. Notez que dans ce mode de fonctionnement, les règles de mappage et de blocage d'authentification de canal ne peuvent pas prendre en compte les résultats de la validation de l'ID utilisateur et du mot de passe.

Par exemple, l'objet d'informations d'authentification par défaut est défini sur **ADOPTCTX (YES)** et l'utilisateur `fred` est connecté. Les deux règles `CHLAUTH` suivantes sont configurées:

```
SET CHLAUTH('MY.CHLAUTH') TYPE(ADDRESSMAP) DESCR('Block all access by default') ADDRESS('*') USERSRC(NOACCESS) ACTION(REPLACE)
```

```
SET CHLAUTH('MY.CHLAUTH') TYPE(USERMAP) DESCR('Allow user bob and force  
CONNAUTH') CLNTUSER('bob') CHCKCLNT(REQUIRED) USERSRC(CHANNEL)
```

La commande suivante est émise, avec l'intention d'authentifier la commande en tant que contexte de sécurité adopté de l'utilisateur bob:

```
runmqsc -c -u bob QMGR
```

En fait, le gestionnaire de files d'attente utilise le contexte de sécurité fred, et non bob, et la connexion échoue.

Pour utiliser le contexte de sécurité de bob, **ChlauthEarlyAdopt** doit être défini sur Y.

PasswordProtection = Compatible (par défaut) |always|optional|warn

Les données d'authentification que les applications IBM MQ client spécifient lorsqu'elles se connectent à un gestionnaire de files d'attente peuvent être protégées à l'aide de la fonction de protection par mot de passe IBM MQ MQCSP, si la connexion n'utilise pas le chiffrement TLS.

La protection par mot de passe MQCSP est utile à des fins de test et de développement car l'utilisation de la protection par mot de passe MQCSP est plus simple que la configuration du chiffrement TLS, mais pas sécurisée.

Pour plus d'informations sur la protection des données d'identification dans la structure MQCSP et sur les valeurs pouvant être définies pour cet attribut, voir [Protection par mot de passe MQCSP](#).

IgnoreSeqNumberMismatch = NO (valeur par défaut) | YES

Les agents MCA (Message Channel Agent) aux deux extrémités d'un canal conservent le nombre de messages envoyés via le canal pour maintenir la synchronisation. La synchronisation peut être perdue, par exemple si la définition de canal à une extrémité est supprimée, puis recrée. Dans ces circonstances, un RESET CHANNEL peut être requis pour reconnaître que des données de synchronisation ont été perdues et permettre au canal de poursuivre le démarrage.

L'attribut **IgnoreSeqNumberMismatch** doit être défini sur le gestionnaire de files d'attente de réception.

En effet, cet attribut exécute une commande de réinitialisation de canal sur le canal récepteur.

Cet attribut contrôle la façon dont le gestionnaire de files d'attente gère une non-concordance de numéro de séquence lors du démarrage du canal à l'aide des valeurs suivantes:

NO

Les numéros de séquence de canal sont vérifiés lors de la resynchronisation du canal. Si les deux agents MCA ne sont pas d'accord sur le même numéro de séquence, le message d'erreur AMQ9526 est signalé et le canal ne démarre pas.

YES

Les numéros de séquence de canal sont vérifiés lors de la resynchronisation du canal, mais si les deux agents MCA ne sont pas d'accord sur le même numéro de séquence, le message d'avertissement AMQ9703 est signalé et le démarrage du canal se poursuit. Cette valeur d'attribut ne doit pas être nécessaire dans des circonstances normales. Lorsqu'il est connu que des données de synchronisation ont été perdues, par exemple lors d'une reprise après incident, cette option évite d'avoir à accuser réception manuellement de chaque non-concordance de numéro de séquence. La spécification de cette valeur a un effet similaire à celui d'un administrateur qui émet automatiquement un **RESET CHANNEL** en réponse à chaque non-concordance de numéro de séquence.

ChlauthIgnoreUserCase = N (valeur par défaut) | Y

Permet à un gestionnaire de files d'attente de rendre le nom d'utilisateur correspondant dans les règles CHLAUTH insensible à la casse. Cette option permet:

- CLNTUSER dans les règles CHLAUTH TYPE (USERMAP) à mettre en correspondance-insensible à la casse
- USERLIST dans les règles CHLAUTH TYPE (BLOCKUSER) à mettre en correspondance avec la casse-insensible

Les valeurs admises pour **ChlauthIgnoreUserCase** sont les suivantes:

N

Les règles d'authentification de canal tentent de faire correspondre l'identification de l'utilisateur client avec la sensibilité à la casse. Par exemple, une règle spécifiant CLNTUSER ('Fred') ne correspondra pas à 'fred' ou 'FRED', elle ne correspondra qu'à un identificateur d'utilisateur de 'Fred'. Il s'agit de la valeur par défaut.

Y

Les règles d'authentification de canal tentent de faire correspondre l'identification de l'utilisateur client avec une insensibilité à la casse. Par exemple, une règle d'authentification de canal avec TYPE (USERMAP) ou TYPE (USERBLOCK) spécifiant CLNTUSER ('Fred') correspond à toute variation de casse, par exemple, les identificateurs d'utilisateur 'Fred', 'FRED' et 'fred' correspondent tous.

Notez que, lorsque vous ignorez la casse des identificateurs utilisateur lors de la mise en correspondance des règles d'authentification de canal, il est possible que plusieurs règles correspondent. Si cela se produit, la règle qui est mise en correspondance n'est pas définie. Par exemple, avec les règles suivantes, si l'utilisateur 'fred' se connecte à un gestionnaire de files d'attente via le canal CLIENT, il peut être mappé à 'mquser1' ou à 'mquser2':

```
SET CHLAUTH('CLIENT') TYPE(USERMAP) CLNTUSER('fred') USERSRC(MAP) MCAUSER('mquser1')
SET CHLAUTH('CLIENT') TYPE(USERMAP) CLNTUSER('FRED') USERSRC(MAP) MCAUSER('mquser2')
```

Pour éviter toute incertitude lors de l'utilisation de ChlauthIgnoreUserCase=Y, évitez de définir des règles CHLAUTH qui se chevaucheraient et qui se traduiraient par un comportement différent lors de l'utilisation d'une correspondance insensible à la casse.

ChlauthIssueWarn = y

Définissez cet attribut si vous souhaitez que le message AMQ9787 soit généré lorsque vous définissez l'attribut WARN = YES sur la commande **SET CHLAUTH**.

Exemple de section

```
Channels:
  MaxChannels=200
  MaxActiveChannels=100
  MQIBindType=STANDARD
  PipelineLength=2
```

Concepts associés

«Etats des canaux», à la page 247

Un canal peut être dans l'un des nombreux états à tout moment. Certains états ont également des sous-états. A partir d'un état donné, un canal peut passer à d'autres états.

Multi

Section de connexion du fichier qm.ini

La section Connexion définit le type de liaison par défaut.

Utilisez la section Connexion du fichier qm.ini pour spécifier le type de liaison par défaut.

Windows

Linux

Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page des propriétés du gestionnaire de files d'attente IBM MQ Explorer Extended.

Remarque : Vous devez créer une section Connexion si vous en avez besoin.

DefaultBindType = SHARED (par défaut) | ISOLÉ

Si **DefaultBindType** est défini sur ISOLÉ, les applications et le gestionnaire de files d'attente s'exécutent dans des processus distincts et aucune ressource n'est partagée entre eux.

Si **DefaultBindType** est défini sur SHARED, les applications et le gestionnaire de files d'attente s'exécutent dans des processus distincts, mais certaines ressources sont partagées entre eux.

La valeur par défaut est SHARED.



Avertissement : DefaultBindType s'applique à tous les appels MQCONN et à toute utilisation de MQCONNX avec MQCNO_STANDARD_BINDING.

La modification de **DefaultBindType** peut entraîner une dégradation des performances de certaines applications.

Exemple de section

```
Connection:
DefaultBindType=SHARED
```

Multi


Consignation des messages de diagnostic

Les journaux de messages de diagnostic de IBM MQ sont un mécanisme qui permet aux différents composants du système IBM MQ de signaler les messages de diagnostic relatifs aux modifications et aux problèmes d'état d'exécution et de configuration de IBM MQ .

Ces journaux sont parfois appelés IBM MQ journaux d'erreurs , mais ils contiennent toujours des messages d'information et d'avertissement IBM MQ , ainsi que des messages d'erreur. Les trois principaux composants de IBM MQ qui rendent compte à ces journaux sont les suivants:

- Gestionnaires de files d'attente
- IBM MQ Clients
- Le reste du système IBM MQ

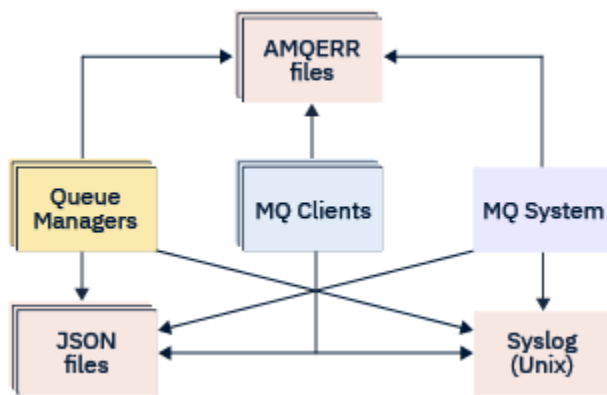
IBM MQ prend en charge la génération de rapports sur les messages de diagnostic via un certain nombre de méthodes différentes appelées *services de message de diagnostic*, permettant une approche personnalisée de l'enregistrement et de la consommation de ces informations:

- Fichiers journaux AMQERRnn
- Fichiers journaux au format JSON
-  Syslog au format JSON

La sortie JSON de IBM MQ est formatée en tant qu'objets JSON à une seule ligne, de sorte que chaque ligne individuelle du journal JSON, ou enregistrement Syslog, représente un objet JSON valide. Le journal dans son ensemble n'est pas encapsulé en tant qu'objet JSON unique.

L'illustration suivante montre que les gestionnaires de files d'attente, les clients IBM MQ et le système IBM MQ peuvent *tous* les messages de diagnostic à l'aide des méthodes décrites.

Figure 5. Comment les différentes parties de IBM MQ peuvent signaler des messages de diagnostic



Configuration des journaux de diagnostic IBM MQ :

Les journaux de diagnostic sont définis et personnalisés à l'aide de strophes du fichier `qm.ini` spécifique au composant IBM MQ qui les requiert. Chaque noeud final de journalisation unique est défini sous son propre en-tête de section dans le fichier ini, ainsi que les personnalisations qui y sont définies. Les personnalisations peuvent inclure:

- Taille des fichiers journaux à encapsuler, avant que la remise à zéro ne se produise ; non applicable à Syslog
- Tout filtrage basé sur la gravité des messages de journal, et
- Tout code de message spécifique à supprimer.

IBM MQ peut être configuré pour écrire dans tout ou partie des trois types de noeud final de journalisation, ce qui permet à des sections de journal particulières de remplir des rôles particuliers. De même, plusieurs services de fichiers peuvent être définis. Exemple :

- Le format JSON facilite l'analyse syntaxique via des outils automatisés dans les environnements locaux et Cloud.
- La sortie syslog permet aux composants IBM MQ d'intégrer des informations de diagnostic dans un emplacement de consignation du système d'exploitation commun en ligne avec les autres produits du système.
- Points de terminaison de journal filtrés en fonction de la gravité permettant à des fichiers journaux particuliers d'enregistrer, par exemple, uniquement les erreurs graves dans le système.

Quel que soit le style de consignation des diagnostics configuré, les fichiers de diagnostic traditionnels stockés dans le répertoire des journaux système IBM MQ (`/var/mqm/errors/AMQERRnn.log`) et dans le répertoire des journaux du gestionnaire de files d'attente spécifique (`/var/mqm/qmgrs/<qmgr_name>/errors/AMQERRnn.log`) sont toujours écrits, en plus de toute autre configuration de consignation utilisée.

Pour les gestionnaires de files d'attente uniquement, la configuration facultative de ces journaux obligatoires peut être effectuée en spécifiant les attributs de «[Sections de service de message de diagnostic](#)», à la page 139.

Différentes zones de strophe

Les strophes supplémentaires peuvent être appliquées à différentes zones de IBM MQ.

Gestionnaire de files d'attente (qm.ini)

S'applique aux messages de journal générés par le gestionnaire de files d'attente

Système (mqs.ini)

S'applique aux messages de journal générés par le système. Cette option n'est pas spécifique à un gestionnaire de files d'attente, sauf lorsqu'un gestionnaire de files d'attente ne peut pas accéder à ses propres journaux ou y écrire des données.

Modèles (mq5.ini)

Une ou plusieurs sections en tant que modèles, qui sont copiées dans `qm.ini` lorsqu'un gestionnaire de files d'attente est créé.

Client (mqclient.ini)

S'applique au fonctionnement du client, par exemple `runmqsc` en mode client sur un gestionnaire de files d'attente éloignées.

Conversion entre les journaux au format JSON et les journaux au format traditionnel

La commande `mqrc` a été améliorée pour permettre un certain nombre de conversions entre les journaux JSON et les journaux au format traditionnel, ainsi qu'entre les différentes langues.

Référence associée

«Sections de service de message de diagnostic», à la page 139

Les options de service de message de diagnostic disponibles permettent la personnalisation de la journalisation des diagnostics IBM MQ, de sorte que la sortie de journal puisse être dirigée vers différents points de terminaison de journal à partir de différents composants d'IBM MQ.

«Section QMErrorLog», à la page 138

Vous utilisez la strophe du journal des erreurs du gestionnaire de files d'attente QMErrorLog dans le fichier `qm.ini` pour personnaliser l'opération et le contenu des journaux des erreurs IBM MQ.

«Services de message de diagnostic», à la page 142

Les services de message de diagnostic suivants et leurs attributs spécifiques, définis dans les sections de modèle DiagnosticSystemMessages, DiagnosticMessages et DiagnosticMessagesde vos fichiers de configuration, peuvent être définis:

Multi Section QMErrorLog

Vous utilisez la strophe du journal des erreurs du gestionnaire de files d'attente QMErrorLog dans le fichier `qm.ini` pour personnaliser l'opération et le contenu des journaux des erreurs IBM MQ.

Le service QMErrorLog est le service de journalisation des diagnostics IBM MQ traditionnel utilisé pour générer des messages de diagnostic relatifs au gestionnaire de files d'attente. Le service QMErrorLog s'exécute en continu et ne peut pas être désactivé, mais peut être personnalisé dans une certaine mesure.

Vous pouvez utiliser la section QMErrorLog du fichier `qm.ini` pour exclure l'écriture de certains messages dans le journal des erreurs du gestionnaire de files d'attente. Vous pouvez également empêcher l'écriture de messages dans le journal des erreurs pendant un intervalle de temps donné.

Windows **Linux** Sinon, au lieu d'éditer directement le fichier `qm.ini`, vous pouvez utiliser la page des propriétés étendues du gestionnaire de files d'attente dans IBM MQ Explorer pour exclure et supprimer des messages avec les attributs **Messages exclus**, **Messages supprimés** et **Intervalle des messages supprimés**.



Avertissement :

- Windows** Vous pouvez utiliser IBM MQ Explorer pour effectuer les modifications uniquement si vous utilisez un gestionnaire de files d'attente local sur la plateforme Windows.
- La section QMErrorLog n'est pas applicable au fichier de configuration du système IBM MQ, `mq5.ini`, ou au fichier de configuration du client, généralement nommé `mqclient.ini`.

Les attributs suivants peuvent être inclus dans la section QMErrorLog :

ErrorLogSize = *taille_max*

Indique la taille du journal des erreurs du gestionnaire de files d'attente qui est copié dans la sauvegarde. `maxsize` doit être compris entre 32768 et 2147483648 octets. Si **ErrorLogSize** n'est pas spécifié, la valeur par défaut de 33554432 octets (32 Mo) est utilisée.

Vous pouvez utiliser cet attribut pour ramener la taille maximale à la taille maximale précédente de 2 Mo, si nécessaire.

Vous pouvez définir la taille du journal à l'aide de la variable d'environnement **MQMAXERRORLOGSIZE** .

ExcludeMessage= *msgIds*

Indique les messages qui ne doivent pas être écrits dans le journal des erreurs du gestionnaire de files d'attente.

Pour plus d'informations, voir [ExcludeMessage](#) dans «Sections de service de message de diagnostic», à la page 139 .

SuppressMessage= *msgIds*

Indique les messages qui sont écrits dans le journal des erreurs du gestionnaire de files d'attente une seule fois dans un intervalle de temps spécifié. Si le même ID de message est spécifié à la fois dans SuppressMessage et dans ExcludeMessage, le message est exclu.

Cette option n'est pas applicable aux services de message de diagnostic définis dans `mqclient.ini`. Pour plus d'informations, voir [SuppressMessage](#) dans «Sections de service de message de diagnostic», à la page 139.

SuppressInterval= *longueur*

Indique l'intervalle de temps, en secondes, au cours duquel les messages spécifiés dans SuppressMessage sont écrits une seule fois dans le journal des erreurs du gestionnaire de files d'attente. La *longueur* doit être comprise entre 1 et 86400 secondes. Si SuppressInterval n'est pas spécifié, la valeur par défaut de 30 secondes est utilisée.

Exemple de section

```
QMErrorLog:
  ErrorLogSize=262144
  ExcludeMessage=7234
  SuppressMessage=9001,9002,9202
  SuppressInterval=30
```

Concepts associés

«Fichiers de configuration du gestionnaire de files d'attente, `qm.ini`», à la page 111

Un fichier de configuration de gestionnaire de files d'attente, `qm.ini`, contient des informations relatives à un gestionnaire de files d'attente spécifique. Les attributs que vous pouvez utiliser pour modifier la configuration d'un gestionnaire de files d'attente individuel remplacent tous les paramètres d' IBM MQ.

Référence associée

«Sections de service de message de diagnostic», à la page 139

Les options de service de message de diagnostic disponibles permettent la personnalisation de la journalisation des diagnostics IBM MQ , de sorte que la sortie de journal puisse être dirigée vers différents points de terminaison de journal à partir de différents composants d' IBM MQ.

Multi Sections de service de message de diagnostic

Les options de service de message de diagnostic disponibles permettent la personnalisation de la journalisation des diagnostics IBM MQ , de sorte que la sortie de journal puisse être dirigée vers différents points de terminaison de journal à partir de différents composants d' IBM MQ.

Vous activez des services de message de diagnostic supplémentaires, à l'aide d'une section portant l'un des noms suivants:

• **DiagnosticSystemMessages**

Définit les services utilisés lors de la génération d'un message de diagnostic qui est consigné dans le journal des erreurs système. Valide dans les fichiers `mqc.ini` ou `mqclient.ini` .

Les applications client utilisent une section **DiagnosticSystemMessages** dans le fichier `mqclient.ini` et dans `mqc.ini`, la section **DiagnosticSystemMessages** contrôle les messages d'une application serveur qui ne possède pas de contexte de gestionnaire de files d'attente.

Il est possible de configurer un gestionnaire de files d'attente et des applications qui écrivent en plus tous les messages dans le service `syslog`.

- **DiagnosticMessages**

Définit les services utilisés lors de la génération d'un message de diagnostic qui est consigné dans le journal des erreurs du gestionnaire de files d'attente. Valide uniquement dans le fichier `qm.ini`.

- **DiagnosticMessagesTemplate**

Section copiée du fichier `mqs.ini` vers **DiagnosticMessages** dans le fichier `qm.ini` lors de la création d'un gestionnaire de files d'attente.

Pour afficher les messages de diagnostic, utilisez la commande `mqrc`.

Attributs des strophes



Avertissement : `Service` et le nom d'une section sont obligatoires.

name= < nom_strophe >

Nom d'une strophe. La valeur doit être unique dans un fichier ini.

Service = type de service

Cet attribut définit un service, dans lequel le nom du service n'est pas sensible à la casse, qui est activé par cette section.

Par exemple, pour activer `syslog` en tant que service supplémentaire, entrez ce qui suit:

```
Service=syslog
```


Voir «Services de message de diagnostic», à la page 142 et leurs attributs spécifiques qui peuvent être utilisés avec les sections du service de message de diagnostic.

Vous pouvez ajouter les attributs facultatifs suivants aux sections:

- [ExcludeMessage](#)
- [SuppressMessage](#)
- [SuppressInterval](#)
- «Gravités», à la page 142

ExcludeMessage= msgIds


Indique les messages qui ne doivent pas être écrits dans le journal des erreurs du gestionnaire de files d'attente. Si votre système IBM MQ est fortement utilisé, avec de nombreux canaux en cours d'arrêt et de démarrage, un grand nombre de messages d'information sont envoyés à la console z/OS et au journal papier. Le pont IBM MQ - IMS et le gestionnaire de mémoire tampon peuvent également produire un grand nombre de messages d'information, de sorte que l'exclusion des messages vous empêche de recevoir un grand nombre de messages si vous en avez besoin. `msgIds` contient une liste d'ID de message séparés par des virgules parmi les suivants:

- 5211-La longueur maximale du nom de propriété a été dépassée.
- 5973-Abonnement de publication / abonnement distribué interdit
- 5974-Publication de publication / abonnement distribuée interdite
- 6254-Le système n'a pas pu charger dynamiquement la bibliothèque partagée
-  7163 - Message de démarrage du travail (IBM i uniquement)
- 7234 - Nombre de messages chargés
- 8245-L'entité ne dispose pas des droits suffisants pour afficher l'objet
- 9001 - Programme de canal correctement terminé
- 9002 - Programme de canal démarré
- 9202 - Hôte éloigné indisponible
- 9208-Erreur lors de la réception de l'hôte
- 9209-Connexion fermée
- 9228-Impossible de démarrer le répondeur de canal

9489-Limite maximale d'instances SVRCONN dépassée
 9490-Nombre maximal d'instances SVRCONN par limite client dépassé
 9508-Impossible de se connecter au gestionnaire de files d'attente
 9524 - Gestionnaire de files d'attente éloignées indisponible
 9528 - Fermeture du canal demandée par l'utilisateur
 9545 -Intervalle de déconnexion expiré
 9558-Le canal distant n'est pas disponible
 9637-Le canal ne possède pas de certificat
 9776-Le canal a été bloqué par l'ID utilisateur
 9777-Le canal a été bloqué par la mappe NOACCESS
 9782-La connexion a été bloquée par l'adresse
 9999 - Fin anormale du programme de canal

SuppressMessage= msgIds

Indique les messages qui sont écrits dans le journal des erreurs du gestionnaire de files d'attente une seule fois dans un intervalle de temps spécifié. Si votre système IBM MQ est fortement utilisé, avec de nombreux canaux en cours d'arrêt et de démarrage, un grand nombre de messages d'information sont envoyés à la console z/OS et au journal papier. Le pont et le gestionnaire de mémoire tampon IBM MQ - IMS peuvent également générer un grand nombre de messages d'information. Par conséquent, la suppression des messages vous empêche de recevoir un certain nombre de messages répétitifs si vous en avez besoin. L'intervalle de temps est spécifié par SuppressInterval. *msgIds* contient une liste d'identificateurs de message séparés par des virgules parmi les suivants:

5211-La longueur maximale du nom de propriété a été dépassée.
 5973-Abonnement de publication / abonnement distribué interdit
 5974-Publication de publication / abonnement distribuée interdite
 6254-Le système n'a pas pu charger dynamiquement la bibliothèque partagée
 7163 - Message de démarrage du travail (IBM i uniquement)
 7234 - Nombre de messages chargés
 8245-L'entité ne dispose pas des droits suffisants pour afficher l'objet
 9001 - Programme de canal correctement terminé
 9002 - Programme de canal démarré
 9202 - Hôte éloigné indisponible
 9208-Erreur lors de la réception de l'hôte
 9209-Connexion fermée
 9228-Impossible de démarrer le répondeur de canal
 9489-Limite maximale d'instances SVRCONN dépassée
 9490-Nombre maximal d'instances SVRCONN par limite client dépassé
 9508-Impossible de se connecter au gestionnaire de files d'attente
 9524 - Gestionnaire de files d'attente éloignées indisponible
 9528 - Fermeture du canal demandée par l'utilisateur
 9545 -Intervalle de déconnexion expiré
 9558-Le canal distant n'est pas disponible
 9637-Le canal ne possède pas de certificat
 9776-Le canal a été bloqué par l'ID utilisateur
 9777-Le canal a été bloqué par la mappe NOACCESS
 9782-La connexion a été bloquée par l'adresse
 9999 - Fin anormale du programme de canal

Si le même ID de message est spécifié à la fois dans SuppressMessage et dans ExcludeMessage, le message est exclu.

Cette option n'est pas applicable aux services de message de diagnostic définis dans MQ client.ini.

SuppressInterval= longueur

Indique l'intervalle de temps, en secondes, pendant lequel les messages spécifiés dans **SuppressMessage** sont écrits une seule fois dans le journal des erreurs du gestionnaire de files d'attente. La *longueur* doit être comprise entre 1 et 86400 secondes. Si **SuppressInterval** n'est pas spécifié, la valeur par défaut de 30 secondes est utilisée.

Gravités

Liste de niveaux de gravité séparés par des virgules, dans laquelle le nom du niveau de gravité n'est pas sensible à la casse. Les valeurs admises sont les suivantes:

- I (ou Information ou 0)
- W (ou Avertissement ou 10)
- E (ou Erreur ou 20 et 30)
- S (ou Stop ou 40)
- T (ou système ou 50)

Remarques :

1. La valeur par défaut est all
2. Seuls les messages des niveaux de gravité sélectionnés sont présentés au service.

Vous pouvez également utiliser le caractère plus (+) qui affiche le niveau d'erreur spécifié et tous les niveaux supérieurs. Par exemple, pour afficher toutes les erreurs:

```
Severities=E+
```

Référence associée

«Section QMErrorLog», à la page 138

Vous utilisez la strophe du journal des erreurs du gestionnaire de files d'attente QMErrorLog dans le fichier `qm.ini` pour personnaliser l'opération et le contenu des journaux des erreurs IBM MQ .

«Services de message de diagnostic», à la page 142

Les services de message de diagnostic suivants et leurs attributs spécifiques, définis dans les sections de modèle DiagnosticSystemMessages, DiagnosticMessages et DiagnosticMessagesde vos fichiers de configuration, peuvent être définis:

Multi *Services de message de diagnostic*

Les services de message de diagnostic suivants et leurs attributs spécifiques, définis dans les sections de modèle DiagnosticSystemMessages, DiagnosticMessages et DiagnosticMessagesde vos fichiers de configuration, peuvent être définis:

Les services de message de diagnostic suivants sont définis:

Fichier

Ce service envoie tous les messages non filtrés à un fichier de la même manière que le service QMErrorLog . Le format textuel existant ou le format JSON spécifié est utilisé en fonction du **Format**spécifié. Par défaut, il existe trois fichiers appelés AMQERR01 . LOG, AMQERR02 . LOGet AMQERR03 . LOG ou AMQERR01 . json, AMQERR02 . jsonet AMQERR03 . json, en fonction de la propriété **Format** , et ces remplacements sont basés sur la taille configurée.



Les attributs suivants sont pris en charge dans une strophe File uniquement:

FilePath

Chemin dans lequel les fichiers journaux sont écrits. Par défaut, il s'agit du même emplacement que les fichiers AMQERR01 . log , c'est-à-dire le système ou le gestionnaire de files d'attente. Le chemin doit être absolu, mais peut inclure des insertions remplaçables. Exemple :



+ MQ_Q_MGR_DATA_PATH +

Chemin d'accès complet au parent du répertoire des messages de diagnostic du gestionnaire de files d'attente. Les valeurs par défaut sont les suivantes :

-  Sur les plateformes AIX and Linux : /var/mqm/qmgrs/<QM_name>
-  sous Windows, C:\Program Data\IBM\MQ\qmgrs\<QM_name>

+ CHEMIN_DATA_MQ +

Chemin d'accès complet au parent du répertoire des messages de diagnostic du système. Les valeurs par défaut sont les suivantes :

-  Sur les plateformes AIX and Linux : /var/mqm
-  Sous Windows : C:\Program Data\IBM\MQ

Vous devez créer ce chemin avec les droits appropriés, s'il n'utilise pas le répertoire des erreurs existant.

FilePrefix

Préfixe des fichiers journaux. La valeur par défaut est AMQERR.

FileSize

Taille à laquelle le journal est repassé. La valeur par défaut est 32MB, comme avec la propriété **ErrorLogSize** de «Section QMErrorLog», à la page 138, qui est sémantiquement identique.

Remarque : La propriété **ErrorLogSize** s'applique uniquement au service de journal des erreurs par défaut et non aux services de diagnostic personnalisés.

Vous pouvez définir la taille du journal à l'aide de la variable d'environnement **MQMAXERRORLOGSIZE**.

Format

Format du fichier. La valeur peut être *text* (pour des services de style QMErrorLog supplémentaires) ou *json*, qui est la valeur par défaut.

Le suffixe du fichier est .LOG ou .json en fonction de la valeur de cet attribut.

Par exemple, éditez le fichier qm.ini du gestionnaire de files d'attente et ajoutez la section suivante:

```
DiagnosticMessages:
  Service = File
  Name = JSONLogs
  Format = json
  FilePrefix = AMQERR
```

Après le redémarrage, le gestionnaire de files d'attente aura des fichiers AMQERR0x.json dans son répertoire ERRORS.

Vous pouvez définir plusieurs services de fichiers. Cela permet la configuration, comme illustré dans les exemples suivants, où les messages de différentes balises sont répartis sur différents ensembles de journaux:

```
DiagnosticMessages:
  Name=ErrorsToFile
  Service=File
  Severities=E+
  FilePrefix=OnlyErrors

DiagnosticMessages:
  Name=NonErrorstoFile
  Service=File
  Severities=1 W
  FilePrefix=Information
```

Journal Syslog

Le service Syslog n'est pas disponible sous Windows ou IBM i

Vous ne pouvez définir qu'un seul service Syslog et le service Syslog envoie tous les messages non filtrés à syslog à l'aide de la spécification des messages de diagnostic au format JSON . Les informations sont ajoutées à syslog dans l'ordre indiqué dans la table, en commençant par msgID et les insertions.

La gravité du message est mappée au niveau syslog de la manière suivante:

Tableau 13. Mappage de la gravité des messages au niveau syslog	
Gravité	Niveau
0	INFOS DE JOURNAL
10	LOG_WARNING
20	ERRE_LOGSE
30	ERRE_LOGSE
40	ALERTE DE JOURNAL
50	ALERTE DE JOURNAL

L'attribut suivant est pris en charge dans une strophe syslog uniquement:

Ident

Définit la valeur **ident** associée aux entrées syslog. La valeur par défaut est *ibm-mq*.

L'exemple suivant montre des messages d'erreur envoyés à Syslog:

```
DiagnosticMessages:
  Name=ErrorsToSyslog
  Ident=mq
  Service=Syslog
  Severities=E+
```

Pour plus d'informations sur les attributs de strophe génériques, voir «[Sections de service de message de diagnostic](#)», à la page 139 .

Remarques :

1. Pour le service File uniquement, vous pouvez avoir plusieurs strophes, chacune ayant un nom différent. Seule la définition, utilisant le nom final dans la séquence, prend effet.
2. Les modifications apportées à la valeur d'une section entrent en vigueur uniquement lorsque le gestionnaire de files d'attente est redémarré.

Multi

Section ExitPath du fichier qm.ini

La section ExitPath indique le chemin des programmes d'exit utilisateur sur le système du gestionnaire de files d'attente.

Utilisez la section ExitPath du fichier qm.ini pour spécifier le chemin des programmes d'exit utilisateur sur le système du gestionnaire de files d'attente.

Windows

Linux

Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page des propriétés du gestionnaire de files d'attente IBM MQ Explorer Exits .

ExitsDefaultPath = chaîne

L'attribut de chemin ExitsDefault indique l'emplacement de:

- Exits de canal 32 bits pour les clients
- Exits de canal 32 bits et exits de conversion de données pour les serveurs
- Fichiers de commutation de chargement XA non qualifiés

ExitsDefaultPath64= chaîne

L'attribut ExitsDefaultPath64 indique l'emplacement de:

- Exits de canal 64 bits pour les clients
- Exits de canal 64 bits et exits de conversion de données pour les serveurs
- Fichiers de commutation de chargement XA non qualifiés

Exemple de section

```
ExitPath:  
  ExitsDefaultPath=/var/mqm/exits  
  ExitsDefaultPath64=/var/mqm/exits64
```

Multi ExitPropertiesStrophe locale du fichier qm.ini

La section locale ExitProperties indique des informations sur les propriétés d'exit d'un gestionnaire de files d'attente.

Utilisez la section locale ExitProperties du fichier qm . ini pour spécifier des informations sur les propriétés d'exit d'un gestionnaire de files d'attente.

Windows **Linux** Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page de propriétés du gestionnaire de files d'attente de cluster IBM MQ Explorer .

Windows Sous Windows , vous pouvez également spécifier ces informations à l'aide de la commande **amqmdain** .

Par défaut, ce paramètre est hérité de l'attribut **CLWLMode** de la section ExitProperties de la configuration à l'échelle de la machine (décrite dans «Section ExitProperties du fichier mqs.ini», à la page 105).

Modifiez ce paramètre uniquement si vous souhaitez configurer ce gestionnaire de files d'attente d'une autre manière. Cette valeur peut être remplacée pour des gestionnaires de files d'attente individuels à l'aide de l'attribut de mode de charge de travail de cluster dans la page des propriétés du gestionnaire de files d'attente de cluster.

Utilisez la section ExitProperties du fichier mqs . ini pour spécifier les options de configuration utilisées par les programmes d'exit du gestionnaire de files d'attente.

Windows **Linux** Vous pouvez également, sous Linux (x86 et x86-64) et Windows, utiliser la page de propriétés IBM MQ Explorer Extended IBM MQ .

CLWLMode = SAFE (par défaut) | FAST

L'exit CLWL (Cluster Workload) vous permet de spécifier la file d'attente de cluster à ouvrir dans le cluster en réponse à un appel MQI (par exemple, MQOPEN, MQPUT). L'exit CLWL s'exécute en mode FAST ou SAFE en fonction de la valeur que vous spécifiez dans l'attribut **CLWLMode** . Si vous omettez l'attribut **CLWLMode** , l'exit de charge de travail de cluster s'exécute en mode SAFE.

SECURISEE

Exécutez l'exit CLWL dans un processus distinct du gestionnaire de files d'attente. Il s'agit de l'option par défaut.

Si un problème se produit avec l'exit CLWL écrit par l'utilisateur lors de l'exécution en mode SAFE, les événements suivants se produisent:

- Le processus serveur CLWL (amqzlw0) échoue.
- Le gestionnaire de files d'attente redémarre le processus serveur CLWL.
- L'erreur vous est signalée dans le journal des erreurs. Si un appel MQI est en cours, vous recevez une notification sous la forme d'un code retour.

L'intégrité du gestionnaire de files d'attente est préservée.

Remarque : L'exécution de l'exit CLWL dans un processus distinct peut affecter les performances.

RAPIDE

Exécutez l'exit de cluster en ligne dans le processus de gestionnaire de files d'attente.

La spécification de cette option améliore les performances en évitant les coûts de changement de processus associés à l'exécution en mode SAFE, mais au détriment de l'intégrité du gestionnaire de files d'attente. Vous ne devriez exécuter l'exit CLWL en mode FAST que si vous êtes convaincu qu'il n'y a aucun problème avec votre exit CLWL, et que vous êtes particulièrement préoccupé par les performances.

Si un problème se produit lorsque l'exit CLWL s'exécute en mode FAST, le gestionnaire de files d'attente échoue et vous risquez de compromettre l'intégrité du gestionnaire de files d'attente.

Exemple de section

```
ExitPropertiesLocal:  
  CLWLMode=SAFE
```

IBM i

Linux

AIX

Section de système de fichiers du fichier qm.ini

La section Système de fichiers indique si les droits définis dans les journaux des erreurs du gestionnaire de files d'attente doivent rester inchangés ou être remplacés par leurs valeurs par défaut.

Les droits d'accès par défaut définis dans les fichiers journaux des erreurs sont censés être utiles dans la plupart des cas et, par conséquent, la plupart des administrateurs IBM MQ n'ont pas besoin de les modifier.

Toutefois, votre administrateur IBM MQ peut vouloir modifier les droits d'accès à ses fichiers journaux d'erreurs, auquel cas il doit définir l'option de section Filesystem **ValidateAuth=Non**, ce qui permet au gestionnaire de files d'attente de ne pas modifier les droits d'accès par la suite.

Le comportement par défaut (sans **ValidateAuth=No**) est que le gestionnaire de files d'attente vérifie les droits d'accès aux fichiers des journaux d'erreurs du gestionnaire de files d'attente et les remplace par leurs valeurs par défaut. Cette vérification peut se produire à tout moment, y compris au cours d'une opération d'arrêt ou de démarrage du gestionnaire de files d'attente.

Exemple de section

```
Filesystem:  
  ValidateAuth=No
```

V 9.4.0

Linux

AIX

Strophe JWKS du fichier mq.ini

Utilisez la section **JWKS** pour indiquer au gestionnaire de files d'attente comment extraire un ensemble de clés publiques qui peuvent valider les jetons signés par cet émetteur.

EndPoint=chaîne

Indique l'URL d'un document JWKS qui fournit les clés publiques requises pour valider les jetons signés par cet émetteur.

Ce serveur spécifié doit fournir une connexion sécurisée (HTTPS) pour extraire les clés. Le gestionnaire de files d'attente doit également avoir été configuré pour établir des connexions HTTP sécurisées. Voir [HTTPSKeyStore](#).

En cas de doute sur la valeur correcte de cet attribut, consultez l'administrateur du serveur d'authentification.

IssuerName=chaîne

Il doit s'agir du **IssuerName** présent dans les jetons d'authentification JWT fournis / signés par cet émetteur.

Vous pouvez l'établir en consultant la documentation de votre serveur d'authentification ou en examinant un exemple de jeton Web JSON (JWT) émis par le service.

UserClaim=chaîne

Fournit le nom d'une revendication (paire clé/valeur dans un jeton d'authentification) que le gestionnaire de files d'attente utilisera comme ID utilisateur IBM MQ lors de l'établissement des droits IBM MQ .

Si des jetons de ce noeud final sont utilisés uniquement pour l'authentification, c'est-à-dire si **ADOPTCTX** est NO, cet attribut est facultatif, sinon il est obligatoire.

Exemple de section

```
JWKS:  
Endpoint=https://myauthserver.example/jwks  
IssuerName=https://myauthserver.example/jwks  
UserClaim=MQUser
```

Information associée

[Configuration d'un gestionnaire de files d'attente pour l'acceptation de jetons d'authentification à l'aide d'un noeud final JWKS](#)

Multi**Section de journal du fichier qm.ini**

La section Journal contient des informations sur la consignation dans un gestionnaire de files d'attente.

Utilisez la section Journal du fichier qm . ini pour spécifier les informations relatives à la consignation sur un gestionnaire de files d'attente.

Windows**Linux**

Sous Linux (x86 et x86-64) et Windows, vous pouvez également utiliser la page de propriétés du IBM MQ Explorer gestionnaire de files d'attente **Log** .

Par défaut, ces paramètres sont hérités des paramètres spécifiés pour les paramètres de journal par défaut du gestionnaire de files d'attente (décrits dans «Section LogDefaults du fichier mq5.ini», à la page 106). Modifiez ces paramètres uniquement si vous souhaitez configurer ce gestionnaire de files d'attente d'une autre manière.

Pour plus d'informations sur le calcul des tailles de journal, voir «Calcul de la taille du journal», à la page 710.

Remarque : Les limites indiquées dans la liste de paramètres suivante sont définies par IBM MQ. Les limites du système d'exploitation peuvent réduire la taille de journal maximale possible.

LogPrimaryFiles = 3 (valeur par défaut) |2-254 (Windows) |2-510 (systèmes AIX and Linux)

Fichiers journaux alloués lors de la création du gestionnaire de files d'attente.

Le nombre minimal de fichiers journaux principaux que vous pouvez avoir est 2 et le nombre maximal est 254 sur les systèmes Windows ou 510 sur les systèmes AIX and Linux . La valeur par défaut est 3.

Le nombre total de fichiers journaux primaires et secondaires ne doit pas dépasser 255 sur les systèmes Windows ou 511 sur les systèmes AIX and Linux et ne doit pas être inférieur à 3.

La valeur est examinée lorsque le gestionnaire de files d'attente est créé ou démarré. Vous pouvez le modifier après la création du gestionnaire de files d'attente. Toutefois, une modification de la valeur n'est pas effective tant que le gestionnaire de files d'attente n'est pas redémarré et que l'effet n'est pas immédiat.

LogSecondaryFiles = 2 (par défaut) |1-253 (Windows) |1-509 (systèmes AIX and Linux)

Fichiers journaux alloués lorsque les fichiers principaux sont épuisés.

Le nombre minimal de fichiers journaux secondaires est 1 et le nombre maximal est 253 sur les systèmes Windows ou 509 sur les systèmes AIX and Linux . Le nombre par défaut est 2.

Le nombre total de fichiers journaux primaires et secondaires ne doit pas dépasser 255 sur les systèmes Windows ou 511 sur les systèmes AIX and Linux et ne doit pas être inférieur à 3.

La valeur est examinée lorsque le gestionnaire de files d'attente est démarré. Vous pouvez modifier cette valeur, mais les modifications ne prennent effet que lorsque le gestionnaire de files d'attente est redémarré, et même dans ce cas, l'effet peut ne pas être immédiat.

LogFilePages = nombre

Les données de journal sont consignées dans une série de fichiers appelée fichiers journaux. La taille de ces fichiers est définie en unité de pages de 4 ko.

Le nombre de pages par défaut d'un fichier journal est égal à 4 096, ce qui signifie que la taille du fichier journal est égale à 16 Mo.

Sur les systèmes AIX and Linux , le nombre minimal de pages de fichier journal est de 64 et sur Windows , le nombre minimal de pages de fichier journal est de 32 ; dans les deux cas, le nombre maximal est de 65 535.

Remarque : La taille des fichiers journaux spécifiés lors de la création du gestionnaire de files d'attente ne peut pas être modifiée pour un gestionnaire de files d'attente.

LogType = CIRCULAIRE (par défaut) | LINEAR | RÉPLIQUÉ

Type de journalisation à utiliser par le gestionnaire de files d'attente. La valeur par défaut est CIRCULAIRE. Pour plus d'informations sur la création d'un gestionnaire de files d'attente avec le type de journalisation requis, voir la description de l'attribut **LogType** dans «[Section LogDefaults du fichier mq.ini](#)», à la page 106.

REUTILISATION

Démarrez la reprise par redémarrage à l'aide du journal pour annuler les transactions qui étaient en cours lorsque le système s'est arrêté.

Pour plus d'informations sur la journalisation avec réutilisation automatique des journaux, voir «[Types de journalisation](#)», à la page 704 .

LINEAR

Pour la reprise par redémarrage et la reprise sur support ou la reprise en aval (création de données perdues ou endommagées en rélisant le contenu du journal).

Voir «[Types de journalisation](#)», à la page 704 pour une explication plus complète de la journalisation linéaire.

CP4I REPLIQUE

Utilisé par un groupe Native HA pour répliquer les données de journal de l'instance active vers les instances de réplique.

Voir «[Types de journalisation](#)», à la page 704 pour une explication plus complète de la journalisation répliquée.

Remarque : Le **LogType** d'un gestionnaire de files d'attente ne peut pas être modifié en modifiant cet attribut dans le fichier `qm.ini` . Pour modifier le **LogType** d'un gestionnaire de files d'attente, vous devez utiliser la commande **migmqlog** .

LogBufferPages=0 (valeur par défaut) |0-4096

Quantité de mémoire allouée aux enregistrements de mémoire tampon pour l'écriture, en spécifiant la taille des mémoires tampon en unités de pages de 4 Ko.

Le nombre minimal de pages tampon est 18 et le nombre maximal est 4096. Les mémoires tampon de taille plus importante améliorent le rendement, en particulier pour les messages plus grands.

Si vous spécifiez 0 (la valeur par défaut), le gestionnaire de files d'attente sélectionne la taille.




Si vous indiquez un nombre compris entre 1 et 17, la valeur par défaut du gestionnaire de files d'attente est 18 (72 Ko). Si vous indiquez un nombre compris entre 18 et 4096, le gestionnaire de files d'attente utilise le nombre spécifié pour définir la quantité de mémoire allouée.

La valeur est examinée lorsque le gestionnaire de files d'attente est démarré. La valeur peut être augmentée ou diminuée dans les limites indiquées. Toutefois, une modification de la valeur n'est effective qu'au prochain démarrage du gestionnaire de files d'attente.

LogPath= nom_répertoire



Répertoire dans lequel résident les fichiers journaux d'un gestionnaire de files d'attente. Il doit exister sur une unité locale sur laquelle le gestionnaire de files d'attente peut écrire et, de préférence, sur une unité différente des files d'attente de messages. La spécification d'une autre unité offre une protection supplémentaire en cas de défaillance du système.

La valeur par défaut est la suivante :

-  C:\ProgramData\IBM\MQ\log dans Windows.
-   /var/mqm/log sur les systèmes AIX and Linux .

Vous pouvez spécifier le nom d'un répertoire dans la commande **crtmqm** à l'aide de l'indicateur **-ld** . Lorsqu'un gestionnaire de files d'attente est créé, un répertoire est également créé sous le répertoire du gestionnaire de files d'attente, qui est utilisé pour stocker les fichiers journaux. Le nom de ce répertoire est basé sur le nom du gestionnaire de files d'attente. Cela permet de s'assurer que le chemin d'accès au fichier journal est unique et qu'il respecte toutes les limitations relatives à la longueur des noms de répertoire.

Si vous ne spécifiez pas **-ld** dans la commande **crtmqm** , la valeur de l'attribut **LogDefaultPath** est utilisée.

  Sur les systèmes AIX and Linux , l'ID utilisateur **mqm** et le groupe **mqm** doivent disposer des droits complets sur les fichiers journaux. Si vous modifiez l'emplacement de ces fichiers, vous devez les attribuer vous-même. Cela n'est pas obligatoire si les fichiers journaux se trouvent dans les emplacements par défaut fournis avec le produit.

LogWriteIntegrity =SingleWrite|DoubleWrite|TripleWrite (valeur par défaut)

Méthode utilisée par le consignateur pour écrire des enregistrements de journal de manière fiable.

TripleWrite (par défaut)

Sachez que vous pouvez sélectionner **DoubleWrite** mais ceci est interprété par le système comme étant **TripleWrite**.

SingleWrite

Vous ne devez utiliser **SingleWrite** que si le système de fichiers et l'unité hébergeant le journal de reprise IBM MQ garantissent explicitement l'atomicité des écritures 4KB .

En cas d'échec de l'écriture d'une page de 4 ko, quelle qu'en soit la raison, les deux seuls états possibles sont l'image d'avant et l'image d'après. Aucun état intermédiaire n'est possible.

Remarque : S'il y a suffisamment d'accès concurrents dans votre charge de travail persistante, la définition d'une valeur autre que la valeur par défaut, **TripleWrite**, présente un avantage potentiel minimal.

Pour plus d'informations, voir [«LogWriteIntegrity-à l'aide de SingleWrite ou de TripleWrite»](#), à la page 150.

LogManagement = Manuel (par défaut) | Automatique | Archivage

Méthode utilisée pour gérer les extensions de journal, manuellement ou par le gestionnaire de files d'attente. La valeur par défaut est **Manuel**.

L'attribut s'applique uniquement lorsque **LogType** est **LINEAR**.

Si vous modifiez la valeur **LogManagement**, la modification n'est pas prise en compte tant que le gestionnaire de files d'attente n'est pas redémarré.

Si une valeur non reconnue est trouvée pour l'attribut, le gestionnaire de files d'attente ne démarre pas tant que la valeur n'est pas corrigée.

 La propriété **LogManagement** n'est pas valide sous IBM i.

Manuel (par défaut)

Vous gérez les extensions de journal manuellement. Si vous spécifiez cette option, le gestionnaire de files d'attente ne réutilise pas et ne supprime pas les domaines réservés aux journaux, même s'ils ne sont plus nécessaires pour la reprise.

Automatique

Les extensions de journal sont gérées automatiquement par le gestionnaire de files d'attente. Si vous spécifiez cette option, le gestionnaire de files d'attente peut réutiliser ou supprimer les domaines réservés aux journaux dès qu'ils ne sont plus nécessaires pour la reprise. L'archivage n'est pas proposé.

Archive

Les extensions de journal sont gérées par le gestionnaire de files d'attente, mais vous devez informer le gestionnaire de files d'attente lorsque l'archivage de chaque extension de journal est terminé.

Si vous spécifiez cette option, le gestionnaire de files d'attente est libre de réutiliser ou de supprimer un domaine réservé aux journaux, dès qu'il a été averti qu'un domaine qui n'est plus nécessaire pour la reprise a été archivé.

Vous effectuez cette notification à l'aide de la commande **RESET QMGR** MQSC ou de la commande PCF [Reset Queue Manager](#) .

Exemple de section

```
Log:
LogPrimaryFiles=3
LogSecondaryFiles=2
LogFilePages=4096
LogType=CIRCULAR
LogBufferPages=0
LogPath=/var/mqm/log/saturn!queue!manager/
```

Remarque : La valeur zéro pour **LogBufferPages** donne une valeur de 512.

LogWriteIntegrity-à l'aide de **SingleWrite** ou de **TripleWrite**

La valeur de l'option **LogWriteIntegrity** , dans la strophe Log du fichier `qm.ini` , détermine l'algorithme utilisé par le consignateur dans IBM MQ pour écrire les enregistrements de journal dans le journal de reprise. Le paramètre par défaut est **TripleWrite** et ce paramètre est sûr dans presque tous les scénarios possibles.

Le paramètre **LogWriteIntegrity** n'a aucun effet, uniquement lorsqu'une page de journal partielle doit être écrite. Pour un gestionnaire de files d'attente avec une quantité raisonnable d'activité simultanée, ce scénario se produit rarement.

SingleWrite

SingleWrite sélectionne un algorithme qui, dans des circonstances très inhabituelles, peut mieux fonctionner que le paramètre **TripleWrite** par défaut. Le paramètre **SingleWrite** est sûr, uniquement si la plateforme de stockage sous-jacente peut absolument garantir dans tous les cas que les pages de 4KB écrites de manière synchrone dans le journal de reprise IBM MQ sont écrites de manière atomique.

Vous ne devez utiliser le paramètre **SingleWrite** que si le système de fichiers ou l'unité hébergeant le journal de reprise IBM MQ garantit explicitement l'atomicité des écritures 4KB . En d'autres termes, lorsqu'une écriture d'une page 4KB échoue pour une raison quelconque, les deux seuls états possibles

doivent être soit l'image antérieure, soit l'image postérieure, et aucun état intermédiaire ne doit être possible. Dans tous les autres cas, vous devez utiliser TripleWrite.

Sur un système disposant d'un accès concurrent suffisant, le gestionnaire de files d'attente écrit uniquement des pages complètes de données de journal, et si un pourcentage élevé de pages complètes est atteint, il n'y a pas de différence de performances significative entre SingleWrite et TripleWrite.

Sur un système avec peu d'accès concurrent, SingleWrite peut présenter un avantage significatif en termes de performances, mais la solution préférée consiste généralement à augmenter l'accès concurrent plutôt qu'à utiliser SingleWrite.

Notez qu'il peut être difficile de déterminer de manière fiable l'atomicité des écritures 4KB et que les modifications apportées au logiciel ou au matériel sous-jacent peuvent invalider une telle garantie.

Si vous doutez que votre infrastructure de stockage offre les garanties requises maintenant et à tout moment dans le futur, vous devez utiliser TripleWrite.

Windows Section LU62 du fichier qm.ini (Windows uniquement)

La section LU62 indique les paramètres de configuration du protocole SNA LU 6.2 . Ces paramètres remplacent les attributs par défaut des canaux.

Utilisez la section LU62 du fichier qm . ini pour spécifier les paramètres de configuration du protocole SNA LU 6.2 . Ils remplacent les attributs par défaut des canaux.

Windows **Linux** Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page des propriétés du gestionnaire de files d'attente IBM MQ Explorer LU6 . 2 .

TPName

Nom du programme transactionnel à démarrer sur le site distant.

Library1= NomDLL1

Nom de la DLL APPC.

La valeur par défaut est WCPIC32.

Library2= DLLName2

Identique à Library1, utilisé si le code est stocké dans deux bibliothèques distinctes.

La valeur par défaut est WCPIC32.

CP4I Section NativeHAInstance du fichier qm.ini

Pour IBM MQ dans des conteneurs, la section NativeHAInstance indique comment les trois noeuds d'une configuration Native HA peuvent communiquer entre eux.

Remarque : Ces informations s'appliquent uniquement aux environnements de conteneur. Voir [Configuration de Native HA à l'aide de l'opérateur IBM MQ](#) ou [Création du groupe Native HA](#) si vous créez vos propres conteneurs.

Vous ajoutez trois sections NativeHAInstance , une pour chaque instance de gestionnaire de files d'attente dans le groupe Native HA (y compris l'instance locale). Ajoutez les attributs suivants:

Nom

Indiquez le nom d'instance que vous avez utilisé lors de la création de l'instance de gestionnaire de files d'attente.

ReplicationAddress

Indiquez le nom d'hôte, IPv4 décimale à point ou IPv6 adresse au format hexadécimal de l'instance. Vous pouvez spécifier l'adresse en tant que nom d'hôte, IPv4 en notation décimale à point ou IPv6 en format hexadécimal. L'adresse de réplication doit pouvoir être résolue et routable à partir de chaque instance du groupe. Le numéro de port à utiliser pour la réplication de journal doit être indiqué entre crochets, par exemple:

```
ReplicationAddress=host1.example.com(4444)
```

Exemple de section

L'exemple suivant illustre la strophe `NativeHAInstance` utilisée dans le fichier `qm.ini` pour spécifier les trois noeuds d'une configuration à haute disponibilité native.

```
NativeHAInstance:  
  Name=node-1  
  ReplicationAddress=host1.example.com(4444)  
NativeHAInstance:  
  Name=node-2  
  ReplicationAddress=host2.example.com(4444)  
NativeHAInstance:  
  Name=node-3  
  ReplicationAddress=host3.example.com(4444)
```

Concepts associés

«Section `NativeHALocalInstance` du fichier `qm.ini`», à la page 152

Pour IBM MQ dans les conteneurs, la section `NativeHALocalInstance` contrôle le fonctionnement d'une configuration Native HA.

CP4I

Section `NativeHALocalInstance` du fichier `qm.ini`

Pour IBM MQ dans les conteneurs, la section `NativeHALocalInstance` contrôle le fonctionnement d'une configuration Native HA.

Remarque : Ces informations s'appliquent uniquement aux environnements de conteneur. Voir [Configuration de Native HA à l'aide de l'opérateur IBM MQ](#) ou [Création du groupe Native HA](#) si vous créez vos propres conteneurs.

La section `NativeHALocalInstance` est ajoutée automatiquement au fichier `qm.ini` sur chacun des noeuds lorsque vous créez une configuration Native HA. Vous pouvez ensuite éditer le fichier `qm.ini` et personnaliser les attributs dans la section `NativeHALocalInstance`.

LocalName

Nom de la section `NativeHALocalInstance`, issu du nom d'instance de réplique de journal spécifié lors de la création du gestionnaire de files d'attente Native HA.

Vous pouvez éventuellement ajouter les attributs suivants à la section `NativeHALocalInstance` :

KeyRepository

Chemin d'accès complet et nom de fichier du référentiel de clés qui contient le certificat numérique utilisé pour protéger le trafic de réplique de journal. Si l'extension de fichier n'est pas spécifiée, elle est supposée être `.kdb`.

Si l'attribut de section `KeyRepository` est omis, les données de réplique de journal sont échangées entre les instances en texte en clair.

V 9.4.0

KeyRepositoryPassword

Le référentiel de clés est sécurisé avec un mot de passe car il contient des informations sensibles. Pour pouvoir accéder au contenu du référentiel de clés, IBM MQ doit pouvoir extraire le mot de passe du référentiel de clés. Si le mot de passe n'est pas stocké dans un fichier de dissimulation de référentiel de clés, vous pouvez le fournir dans l'attribut `KeyRepositoryPassword`. Exemple :

```
KeyRepositoryPassword=passw0rd
```



Avertissement : Si vous indiquez le mot de passe à l'aide de cet attribut, chiffrez-le avec le système de protection par mot de passe IBM MQ. Pour plus d'informations, voir «[Chiffrement du mot de passe du référentiel de clés](#)», à la page 154.

V 9.4.0

InitialKeyFile

Spécifiez cet attribut si le mot de passe du référentiel de clés spécifié avec l'attribut `KeyRepositoryPassword` est chiffré avec une clé initiale spécifique. Le nom du fichier qui contient la clé initiale peut être spécifié à l'aide du paramètre **-sf** lorsque la commande **runmqicred** est utilisée pour chiffrer le mot de passe du référentiel de clés.

Définissez la valeur de cet attribut sur le nom du fichier qui contient la clé initiale utilisée pour chiffrer le mot de passe. Par exemple, si un fichier appelé `mykey.key` contient la clé initiale:

```
InitialKeyFile=/mykey.key
```

Pour plus d'informations, voir «Chiffrement du mot de passe du référentiel de clés», à la page 154.

CertificateLabel

Libellé de certificat identifiant le certificat numérique à utiliser pour la protection du trafic de réplication des journaux. Si `KeyRepository` est fourni mais que `CertificateLabel` est omis, la valeur par défaut `ibmwebspheremqqueue_manager` est utilisée.

CipherSpec

`CipherSpec` à utiliser pour protéger le trafic de réplication de journal. Si cet attribut de section est fourni, `KeyRepository` doit également être fourni. Si `KeyRepository` est fourni mais que `CipherSpec` est omis, la valeur par défaut ANY est utilisée.

LocalAddress

Adresse de l'interface réseau locale qui accepte le trafic de réplication de journal. Si cet attribut de section est fourni, il identifie l'interface réseau locale et / ou le port en utilisant le format "[addr] [(port)]". L'adresse réseau peut être spécifiée sous la forme d'un nom d'hôte, IPv4 à notation décimale à point ou IPv6 au format hexadécimal. Si cet attribut est omis, le gestionnaire de files d'attente tente de se connecter à toutes les interfaces réseau et utilise le port spécifié dans `ReplicationAddress` dans la section `NativeHAInstances` correspondant au nom de l'instance locale.

HeartbeatInterval

L'intervalle des pulsations définit la fréquence en millisecondes à laquelle une instance active d'un gestionnaire de files d'attente Native HA envoie une pulsation réseau. Il est compris entre 500 (0,5 secondes) et 60000 (1 minute). Une valeur hors de cette plage empêche le démarrage du gestionnaire de files d'attente. Si cet attribut est omis, une valeur par défaut de 5000 (5 secondes) est utilisée. Chaque instance doit utiliser le même intervalle de pulsations.

HeartbeatTimeout

Le dépassement du délai d'attente du signal de présence définit le temps pendant lequel une réplique d'instance d'un gestionnaire de files d'attente Native HA attend avant de considérer que l'instance active ne répondra pas. Cette valeur doit être comprise entre 500 (0,5 secondes) et 120000 (2 minutes). La valeur du dépassement du délai d'attente du signal de présence doit être supérieure ou égale à celle de l'intervalle des pulsations.

Une valeur non valide empêche le démarrage du gestionnaire de files d'attente. Si cet attribut est omis, une réplique attend 2 x `HeartbeatInterval` avant de lancer le processus pour sélectionner une nouvelle instance active. Chaque instance doit utiliser la même valeur de dépassement du délai d'attente du signal de présence.

RetryInterval

L'intervalle entre les nouvelles tentatives définit la fréquence en millisecondes à laquelle un gestionnaire de files d'attente Native HA doit retenter un lien de réplication défectueux. Cet intervalle doit être compris entre 500 (0,5 secondes) et 120000 (2 minutes). Si cet attribut est omis, une réplique attend 2 x `HeartbeatInterval` avant de réessayer un lien de réplication ayant échoué.

SSLFipsRequired

Indique si seuls les algorithmes certifiés FIPS sont utilisés si la cryptographie est utilisée lors de l'envoi du trafic de réplication de journal. Définissez la valeur sur Yes ou No.

EncryptionPolicySuiteB

Indique si le trafic de réplication de journal utilise la cryptographie compatible Suite-B et quel niveau de puissance est utilisé. Définissez l'une des valeurs suivantes:

NONE

La cryptographie compatible Suite-B n'est pas utilisée. Il s'agit du paramètre par défaut.

128_BIT, 192_BIT

Définit la force de sécurité sur les niveaux 128 bits et 192 bits.

128_BIT

Définit le niveau de sécurité à 128 bits.

192_BIT

Définit le niveau de sécurité sur 192 bits.

V 9.4.0 CompressionThreshold

Définit un seuil d'octets qui, lorsqu'il est franchi, déclenche la compression des données de journal. Les données de journal supérieures à la valeur de seuil sont compressées. La valeur 0 (valeur par défaut supposée) désactive toute compression et la valeur 1 permet de compresser chaque ajout de journal. La valeur maximale est 268435456 (256 Mo).

V 9.4.0 LZ4Acceleration

Paramètre d'optimisation qui contrôle la façon dont l'algorithme LZ4 recherche les séquences compressibles dans les données de journal. Chaque fois que la valeur de l'accélération augmente de 1, l'algorithme n'est pas aussi prudent pour une séquence compressible, échangeant le taux de compression pour un faible gain de performance. La valeur minimale (et la valeur par défaut) est 1 et la valeur maximale est 65537.

Chiffrement du mot de passe du référentiel de clés

V 9.4.0

Le mot de passe du référentiel de clés peut être protégé à l'aide du système de protection par mot de passe IBM MQ ou d'un fichier de dissimulation de référentiel de clés. Pour plus d'informations sur ces deux méthodes, voir [Chiffrement des mots de passe du référentiel de clés](#).

Si le mot de passe du référentiel est spécifié à l'aide de l'attribut `KeyRepositoryPassword` dans la section `NativeHALocalInstance`, chiffrez le mot de passe à l'aide du système de protection par mot de passe IBM MQ. Utilisez la commande **runmqicred** pour chiffrer le mot de passe. La commande renvoie le mot de passe chiffré qui peut être spécifié dans l'attribut `KeyRepositoryPassword`.

Utilisez une clé initiale unique pour chiffrer le mot de passe de manière sécurisée. Le nom du fichier qui contient la clé initiale peut être spécifié à l'aide du paramètre **-sf** dans la commande **runmqicred**. Si vous ne fournissez pas de clé unique, la clé par défaut est utilisée.

Si vous chiffrez le mot de passe du référentiel de clés avec une clé initiale unique, vous devez également fournir la même clé initiale à l'aide de l'attribut `InitialKeyFile` dans la section `NativeHALocalInstance`.

Exemple de section

L'exemple suivant illustre la section `NativeHALocalInstance` utilisée dans le fichier `qm.ini` pour spécifier le nom local d'un noeud.

```
NativeHALocalInstance:  
LocalName=node-1
```

Concepts associés

«Section `NativeHAInstance` du fichier `qm.ini`», à la page 151

Pour IBM MQ dans des conteneurs, la section `NativeHAInstance` indique comment les trois noeuds d'une configuration Native HA peuvent communiquer entre eux.

Référence associée

[runmqicred \(protection des mots de passe du client IBM MQ\)](#)

Windows Section NETBIOS du fichier qm.ini (Windows uniquement)

La section NETBIOS du fichier `qm.ini` spécifie les paramètres de configuration du protocole NetBIOS. Ces paramètres remplacent les attributs par défaut des canaux.

Utilisez la strophe NETBIOS du fichier `qm.ini` pour spécifier les paramètres de configuration du protocole NetBIOS. Ils remplacent les attributs par défaut des canaux.

Windows Linux Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page de propriétés du gestionnaire de files d'attente Netbios IBM MQ Explorer.

LocalName= *nom*

Nom sous lequel cette machine est connue sur le réseau local.

AdapterNum = 0 (valeur par défaut) | *numéro_adaptateur*

Numéro de la carte de réseau local. La valeur par défaut est l'adaptateur 0.

NumSess = 1 (valeur par défaut) | *nombre_de_sessions*

Nombre de sessions à allouer. La valeur par défaut est 1.

NumCmds = 1 (par défaut) | *nombre_de_commandes*

Nombre de commandes à allouer. La valeur par défaut est 1.

NumNames = 1 (valeur par défaut) | *nombre_de_noms*

Nombre de noms à allouer. La valeur par défaut est 1.

Library1= *DLLName1*

Nom de la DLL NetBIOS.

La valeur par défaut est NETAPI32.

Concepts associés

«Définition du nom NetBIOS IBM MQ local», à la page 284

Le nom NetBIOS local utilisé par les processus de canal IBM MQ peut être spécifié de trois manières.

Linux AIX Section RestrictedMode du fichier qm.ini

La section RestrictedMode indique le nom du groupe qui contient les membres autorisés à exécuter des applications MQI, à mettre à jour toutes les ressources du GIEC et à modifier le contenu de certains répertoires du gestionnaire de files d'attente. Cette section s'applique uniquement aux systèmes AIX and Linux.

La section RestrictedMode est définie par l'option **-g** dans la commande `crtmqm`. Si vous n'utilisez pas l'option **-g**, la section n'est pas créée dans le fichier `qm.ini`.

Il existe des répertoires dans lesquels les applications IBM MQ créent des fichiers alors qu'elles sont connectées au gestionnaire de files d'attente dans le répertoire de données du gestionnaire de files d'attente. Pour que les applications puissent créer des fichiers dans ces répertoires, elles disposent d'un accès en écriture à l'échelle mondiale:

- `/var/mqm/sockets/QMgrName/@ipcc/ssem/hostname/`
- `/var/mqm/sockets/QMgrName/@app/ssem/hostname/`
- `/var/mqm/sockets/QMgrName/zsocketapp/hostname/`

où `QMGRNAME` est le nom du gestionnaire de files d'attente et `hostname` est le nom d'hôte.

Sur certains systèmes, il est inacceptable d'accorder à tous les utilisateurs un accès en écriture à ces répertoires. Par exemple, les utilisateurs qui n'ont pas besoin d'accéder au gestionnaire de files d'attente. Le mode restreint modifie les droits des répertoires qui stockent les données du gestionnaire de files d'attente. Les répertoires ne sont alors accessibles qu'aux membres du groupe d'applications spécifié. Les droits d'accès à la mémoire partagée System V IPC utilisés pour communiquer avec le gestionnaire de files d'attente sont également modifiés de la même manière.

Le groupe d'applications est le nom du groupe dont les membres sont autorisés à effectuer les opérations suivantes:

- Exécuter des applications MQI
- Mettre à jour toutes les ressources du GIEC
- Modification du contenu de certains répertoires de gestionnaire de files d'attente

Pour utiliser le mode restreint pour un gestionnaire de files d'attente:

- Le créateur du gestionnaire de files d'attente doit se trouver dans le groupe mqm et dans le groupe d'applications.
- L'ID utilisateur mqm doit figurer dans le groupe d'applications.
- Tous les utilisateurs qui souhaitent administrer le gestionnaire de files d'attente doivent se trouver dans le groupe mqm et dans le groupe d'applications.
- Tous les utilisateurs qui souhaitent exécuter des applications IBM MQ doivent se trouver dans le groupe d'applications.

Tout appel MQCONN ou MQCONNX émis par un utilisateur qui ne fait pas partie du groupe d'applications échoue avec le code anomalie MQRC_Q_MGR_NOT_AVAILABLE.

Important : Sur de nombreux systèmes d'exploitation, pour que l'ajout d'un utilisateur à un groupe soit reconnu, l'utilisateur en question doit se déconnecter et se reconnecter.

Le mode restreint fonctionne avec le service d'autorisation IBM MQ . Par conséquent, vous devez également accorder aux utilisateurs le droit de se connecter à IBM MQ et d'accéder aux ressources dont ils ont besoin à l'aide du service d'autorisation IBM MQ .

ALW Pour plus d'informations sur la configuration du service d'autorisation IBM MQ , voir [Configuration de la sécurité sur les systèmes AIX, Linux, and Windows](#).

Utilisez uniquement le mode restreint IBM MQ lorsque le contrôle fourni par le service d'autorisation ne permet pas d'isoler suffisamment les ressources du gestionnaire de files d'attente.

Référence associée

crtmqm (création d'un gestionnaire de files d'attente)

Multi Section de sécurité du fichier qm.ini

La section Sécurité spécifie des options pour le gestionnaire des droits d'accès aux objets (OAM).

ClusterQueueAccessControl= RQMName | Xmitq

Définissez cet attribut pour vérifier le contrôle d'accès des files d'attente de cluster ou des files d'attente complètes hébergées sur les gestionnaires de files d'attente de cluster.

RQMNAME

Les profils vérifiés pour le contrôle d'accès des files d'attente hébergées à distance sont des files d'attente nommées ou des profils de gestionnaire de files d'attente nommés.

XMITQ

Les profils vérifiés pour le contrôle d'accès des files d'attente hébergées à distance sont résolus dans SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Xmitq est la valeur utilisée par défaut.

Windows GroupModel=GlobalGroups

Cet attribut détermine si la méthode d'accès aux objets (OAM) vérifie les groupes globaux lors de la détermination de l'appartenance à un groupe d'un utilisateur sous Windows.

La valeur par défaut est de ne pas vérifier les groupes globaux.

GlobalGroups

La méthode d'accès aux objets (OAM) vérifie les groupes globaux.

Avec GlobalGroups défini, les commandes d'autorisation, **setmqaut**, **dspmqaut** et **dmpmqaut** acceptent les noms de groupes globaux ; voir le paramètre **setmqaut -g** .

Remarque : Si vous définissez `ClusterQueueAccessControl=RQMName` et que vous utilisez une implémentation personnalisée du service d'autorisation inférieure à `MQZAS_VERSION_6`, le gestionnaire de files d'attente ne démarre pas. Dans cette instance, définissez `ClusterQueueAccessControl=Xmitq` ou mettez à niveau le service d'autorisation personnalisé vers `MQZAS_VERSION_6` ou une version ultérieure.

Exemple de section

```
Security:
  ClusterQueueAccessControl=Xmitq
  GroupModel=GlobalGroups
```

Multi

Section de service du fichier `qm.ini`

La section `Service` est utilisée pour apporter des modifications aux services installables. Cette section contient le nom du service et le nombre de points d'entrée définis pour le service.

Remarque : Windows Linux La modification des services installables et de leurs composants a des implications importantes. Pour cette raison, les services installables sont en lecture seule dans IBM MQ Explorer.

Pour chaque composant d'un service, vous devez également spécifier le nom et le chemin du module contenant le code de ce composant. Pour cela, utilisez la section [ServiceComponent](#).

Les sections **Service** et **ServiceComponent** peuvent apparaître dans n'importe quel ordre et les clés de section qui leur sont associées peuvent également apparaître dans n'importe quel ordre. Pour l'une ou l'autre de ces sections, toutes les clés de section doivent être présentes. Si une clé de section est dupliquée, la dernière est utilisée.

Au démarrage, le gestionnaire de files d'attente traite chaque entrée de composant de service dans le fichier de configuration. Il charge ensuite le module de composant spécifié, en appelant le point d'entrée du composant (qui doit être le point d'entrée pour l'initialisation du composant), en lui transmettant un descripteur de configuration.

Name = AuthorizationService (par défaut) |NameService

Nom du service requis.

AuthorizationService

Pour IBM MQ, le composant **AuthorizationService** est appelé gestionnaire des droits d'accès aux objets ou OAM. La section **Service** et la section **ServiceComponent** qui lui est associée sont ajoutées automatiquement lors de la création du gestionnaire de files d'attente, mais elles peuvent être remplacées par la variable d'environnement `MQSNOAUT`. Ajoutez d'autres sections **ServiceComponent** manuellement.

Linux

AIX

Les exemples suivants de strophes du fichier `qm.ini` définissent deux composants de service d'autorisation sous IBM MQ for AIX. `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé.

```
Service:
  Name=AuthorizationService
  EntryPoints=13

ServiceComponent:
  Service=AuthorizationService
  Name=MQSeries.UNIX.auth.service
  Module=MQ_INSTALLATION_PATH/lib/amqzfu
  ComponentDataSize=0

ServiceComponent:
  Service=AuthorizationService
  Name=user.defined.authorization.service
```

```
Module=/usr/bin/udas01
ComponentDataSize=96
```

Linux AIX

La section `ServiceComponent MQSeries.UNIX.auth.service` définit le composant de service d'autorisation par défaut, OAM. Si vous supprimez cette section et redémarrez le gestionnaire de files d'attente, la méthode d'accès aux objets (OAM) est désactivée et aucune vérification d'autorisation n'est effectuée.

Windows

Vous pouvez également ajouter l'attribut **SecurityPolicy** à l'aide des services IBM MQ. L'attribut **SecurityPolicy** s'applique uniquement si le service spécifié dans la strophe `Service` est le service d'autorisation, c'est-à-dire le service OAM par défaut. L'attribut **SecurityPolicy** permet de spécifier la règle de sécurité pour chaque gestionnaire de files d'attente. Les valeurs possibles sont les suivantes:

Par défaut

Indiquez `Par` par défaut si vous souhaitez que la stratégie de sécurité par défaut soit appliquée. Si un ID de sécurité Windows (ID de sécurité NT) n'est pas transmis à la méthode d'accès aux objets (OAM) pour un ID utilisateur particulier, une tentative est effectuée pour obtenir l'ID de sécurité approprié en effectuant une recherche dans les bases de données de sécurité appropriées.

NTSIDsRequired

Requiert qu'un SID NT soit transmis à la méthode d'accès aux objets (OAM) lors de l'exécution de contrôles de sécurité.

Windows

La section `ServiceComponent MQSeries.WindowsNT.auth.service` définit le composant de service d'autorisation par défaut, OAM. Si vous supprimez cette section et redémarrez le gestionnaire de files d'attente, la méthode d'accès aux objets (OAM) est désactivée et aucune vérification d'autorisation n'est effectuée.

NameService

Aucun service de nom n'est fourni par défaut. Si vous avez besoin d'un service annuaire, vous devez ajouter la section `NameService` manuellement.

Linux AIX

Les exemples suivants de sections de fichier AIX and Linux `qm.ini` pour le service annuaire indiquent un composant de service annuaire fourni par la société ABC (fictive).

```
# Stanza for name service
Service:
  Name=NameService
  EntryPoints=5

# Stanza for name service component, provided by ABC
ServiceComponent:
  Service=NameService
  Name=ABC.Name.Service
  Module=/usr/lib/abcname
  ComponentDataSize=1024
```

Remarque : Windows Sur les systèmes Windows, les informations de section `NameService` sont stockées dans le registre.

EntryPoints= nombre-d'-entrées

Nombre de points d'entrée définis pour le service.

Cela inclut les points d'entrée d'initialisation et de fin.

Windows

SecurityPolicy= Valeur par défaut |NTSIDsRequired

Sur les systèmes Windows, l'attribut **SecurityPolicy** s'applique uniquement si le service spécifié est le service d'autorisation par défaut, c'est-à-dire le service OAM. L'attribut **SecurityPolicy** permet de spécifier la règle de sécurité pour chaque gestionnaire de files d'attente.

Les valeurs possibles sont les suivantes:

Par défaut

Utilisez la stratégie de sécurité par défaut pour prendre effet. Si un ID de sécurité Windows (ID de sécurité NT) n'est pas transmis à la méthode d'accès aux objets (OAM) pour un ID utilisateur particulier, une tentative est effectuée pour obtenir l'ID de sécurité approprié en effectuant une recherche dans les bases de données de sécurité appropriées.

NTSIDsRequired

Transmettez un SID NT à la méthode d'accès aux objets (OAM) lors des contrôles de sécurité.

Pour plus d'informations, voir [Identificateurs de sécurité Windows \(SID\)](#).

Linux AIX SecurityPolicy=utilisateur|groupe|UserExternal|par défaut

Sur les systèmes AIX and Linux, la valeur indique si le gestionnaire de files d'attente utilise une autorisation basée sur l'utilisateur ou sur le groupe. Les valeurs ne sont pas sensibles à la casse.

Les valeurs admises sont les suivantes :

groupe

Le gestionnaire de files d'attente utilise une autorisation basée sur un groupe. Les droits d'accès à une ressource sont accordés à un groupe.

Un utilisateur reçoit l'agrégat de tous les droits accordés à chaque groupe auquel il appartient.

Les ID utilisateur et les groupes doivent être définis sur le système d'exploitation local.

utilisateur

Le gestionnaire de files d'attente utilise l'autorisation basée sur l'utilisateur. Les droits d'accès à une ressource peuvent être accordés à un groupe ou à un ID utilisateur spécifique.

Un utilisateur reçoit l'ensemble des droits suivants:

- Droits accordés à l'utilisateur spécifique.
- Droits accordés à chaque groupe auquel appartient l'utilisateur.

Les ID utilisateur et les groupes doivent être définis sur le système d'exploitation local.

UserExternal

Le gestionnaire de files d'attente utilise l'autorisation basée sur l'utilisateur. Toutefois, des droits peuvent être accordés à des ID utilisateur qui ne sont pas connus du système d'exploitation local.

Les droits d'accès à une ressource peuvent être accordés à un groupe ou à un ID utilisateur spécifique.

Un utilisateur reçoit l'ensemble des droits suivants:

- Droits accordés à l'utilisateur spécifique.
- Droits accordés à chaque groupe auquel appartient l'utilisateur.

Si un utilisateur n'est pas connu du système d'exploitation local, il est considéré comme appartenant uniquement au groupe "personne". Pour plus d'informations sur les groupes, voir [Principaux et groupes sur AIX, Linux, and Windows](#). L'ID utilisateur doit comporter jusqu'à 12 caractères et doit être conforme aux [règles de dénomination des objets IBM MQ](#).

Vous pouvez modifier les gestionnaires de files d'attente existants pour utiliser cette option supplémentaire sans perdre la configuration en cours.

V9.4.0 Il s'agit de la valeur par défaut si la section AuthToken est spécifiée.

par défaut

Le gestionnaire de files d'attente utilise une autorisation basée sur un groupe. Le comportement est identique à celui de l'option group.

Il s'agit de la valeur par défaut si la section AuthToken n'est pas spécifiée.

Redémarrez le gestionnaire de files d'attente pour que les modifications apportées à la valeur d'attribut soient prises en compte.

Remarque : V 9.4.0 Linux AIX Depuis la IBM MQ 9.4.0, si la section AuthToken est spécifiée, la valeur effective de l'attribut **SecurityPolicy** de la section Service est définie sur UserExternal. L'authentification par jeton n'est pas disponible si **SecurityPolicy** est explicitement défini sur Group dans la section Service. Si **SecurityPolicy** est défini sur Groupe, supprimez l'attribut **SecurityPolicy** de la strophe Service, puis redémarrez le gestionnaire de files d'attente. Pour plus d'informations, voir la section «Section AuthToken du fichier qm.ini», à la page 126.

SharedBindingsUserId= type d'utilisateur

L'attribut **SharedBindingsUserId** s'applique uniquement si le service spécifié est le service d'autorisation par défaut, c'est-à-dire le service OAM. L'attribut **SharedBindingsUserId** est utilisé uniquement en relation avec les liaisons partagées. Cette valeur permet de spécifier si la zone *UserIdentifier* de la structure *IdentityContext*, à partir de la fonction MQZ_AUTHENTICATE_USER, est l'ID utilisateur effectif ou l'ID utilisateur réel.

Pour plus d'informations sur la fonction MQZ_AUTHENTICATE_USER, voir [MQZ_AUTHENTICATE_USER-Authenticate user](#).

Les valeurs possibles sont les suivantes:

Par défaut

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur réel.

Réal

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur réel.

A partir du

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur effectif.

FastpathBindingsUserId= type d'utilisateur

L'attribut **FastpathBindingsUserId** s'applique uniquement si le service spécifié est le service d'autorisation par défaut, c'est-à-dire le service OAM. L'attribut **FastpathBindingsUserId** est utilisé avec la relation aux liaisons de raccourci uniquement. Cette valeur permet de spécifier si la zone *UserIdentifier* de la structure *IdentityContext*, à partir de la fonction MQZ_AUTHENTICATE_USER, est l'ID utilisateur effectif ou l'ID utilisateur réel.

Pour plus d'informations sur la fonction MQZ_AUTHENTICATE_USER, voir [MQZ_AUTHENTICATE_USER-Authenticate user](#).

Les valeurs possibles sont les suivantes:

Par défaut

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur réel.

Réal

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur réel.

A partir du

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur effectif.

IsolatedBindingsUserId= type-utilisateur

L'attribut **IsolatedBindingsUserId** s'applique uniquement si le service spécifié est le service d'autorisation par défaut, c'est-à-dire le service OAM. L'attribut **IsolatedBindingsUserId** est utilisé uniquement en relation avec les liaisons isolées. Cette valeur permet de spécifier si la zone *UserIdentifier* de la structure *IdentityContext*, à partir de la fonction MQZ_AUTHENTICATE_USER, est l'ID utilisateur effectif ou l'ID utilisateur réel.

Pour plus d'informations sur la fonction MQZ_AUTHENTICATE_USER, voir [MQZ_AUTHENTICATE_USER-Authenticate user](#).

Les valeurs possibles sont les suivantes:

Par défaut

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur effectif.

Réal

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur réel.

A partir du

La valeur de la zone *UserIdentifier* est définie en tant qu'ID utilisateur effectif.

Pour plus d'informations sur les services et composants installables, voir [Services et composants installables pour AIX, Linux, and Windows](#).

Pour plus d'informations sur les services de sécurité en général, voir [Configuration de la sécurité sur les systèmes AIX and Linux](#).

Exemple de section

```
Service:  
  Name=AuthorizationService  
  EntryPoints=14
```

Concepts associés

[Services et composants installables pour AIX, Linux et Windows](#)

Référence associée

[Services et composants installables sur IBM i](#)

[Informations de référence sur les services installables](#)

Section ServiceComponent du fichier qm.ini

La section ServiceComponent spécifie les informations relatives au composant de service. Vous devez spécifier les informations de composant de service lorsque vous ajoutez un nouveau service installable. La section du service d'autorisation est présente par défaut et le composant associé, OAM, est actif.

Les sections **Service** et **ServiceComponent** peuvent apparaître dans n'importe quel ordre et les clés de section qui leur sont associées peuvent également apparaître dans n'importe quel ordre. Pour l'une ou l'autre de ces sections, toutes les clés de section doivent être présentes. Si une clé de section est dupliquée, la dernière est utilisée.

Au démarrage, le gestionnaire de files d'attente traite chaque entrée de composant de service dans le fichier de configuration. Il charge ensuite le module de composant spécifié, en appelant le point d'entrée du composant (qui doit être le point d'entrée pour l'initialisation du composant), en lui transmettant un descripteur de configuration.

Service = *nom_service*

Nom du service requis. Cette valeur doit correspondre à la valeur spécifiée dans l'attribut Name des informations de configuration du service.

Name = *nom_composant*

Nom descriptif du composant de service. Il doit être unique et contenir uniquement des caractères valides pour les noms d'objets IBM MQ (par exemple, les noms de file d'attente). Ce nom apparaît dans les messages de l'opérateur générés par le service. Nous recommandons que ce nom commence par une marque de la société ou une chaîne distinctive similaire.

Module = *nom_module*

Nom du module devant contenir le code de ce composant. Il doit s'agir d'un nom de chemin d'accès complet.

ComponentDataTaille = *taille*

Taille, en octets, de la zone de données de composant transmise au composant lors de chaque appel. Indiquez zéro si aucune donnée de composant n'est requise.

Exemple de section

```
ServiceComponent:  
  Service=AuthorizationService  
  Name=MQSeries.UNIX.auth.service
```

Pour d'autres exemples illustrant une section AuthorizationService et ses sections ServiceComponent et NameService et sa section ServiceComponent associées, voir «Section de service du fichier qm.ini», à la page 157.

Concepts associés

[Services et composants installables pour AIX, Linux et Windows](#)

Référence associée

«Section de service du fichier qm.ini», à la page 157

La section Service est utilisée pour apporter des modifications aux services installables. Cette section contient le nom du service et le nombre de points d'entrée définis pour le service.

[Services et composants installables sur IBM i](#)

[Informations de référence sur les services installables](#)

Windows Strophe SPX du fichier qm.ini (Windows uniquement)

La section SPX spécifie les paramètres de configuration du protocole SPX. Ces paramètres remplacent les attributs par défaut des canaux.

Utilisez la strophe SPX du fichier qm.ini pour spécifier les paramètres de configuration du protocole SPX.

Windows **Linux** Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la page des propriétés du gestionnaire de files d'attente IBM MQ Explorer SPX.

Socket = 5E86 (valeur par défaut) | numéro_socket

Nombre de sockets SPX en notation hexadécimale. La valeur par défaut est X'5E86'.

BoardNum = 0 (valeur par défaut) | numéro_adaptateur

Numéro de carte de réseau local. La valeur par défaut est l'adaptateur 0.

KeepAlive= NON | OUI

Activez ou désactivez la fonction KeepAlive.

KeepAlive=YES permet à SPX de vérifier périodiquement que l'autre extrémité de la connexion est toujours disponible. Si tel n'est pas le cas, le canal est fermé.

Library1= DLLName1

Nom de la DLL SPX.

La valeur par défaut est WSOCK32.DLL.

Library2= DLLName2

Identique à LibraryName1, utilisé si le code est stocké dans deux bibliothèques distinctes.

La valeur par défaut est WSOCK32.DLL.

ListenerBacklog= nombre

Remplacez le nombre par défaut de demandes en attente pour le programme d'écoute SPX.

Lors de la réception sur SPX, un nombre maximal de demandes de connexion en attente est défini. Il peut s'agir d'un arriéré de demandes en attente sur le socket SPX pour que le programme d'écoute accepte la demande. Les valeurs de journal des éléments en attente du programme d'écoute par défaut sont indiquées dans le [Tableau 14](#), à la page 162.

Plateforme	Valeur ListenerBacklog par défaut
WindowsServeur	100
Windows Poste de travail	5

Remarque : Certains systèmes d'exploitation prennent en charge une valeur supérieure à la valeur par défaut affichée. Utilisez cette option pour éviter d'atteindre le nombre maximal de connexions.

A l'inverse, certains systèmes d'exploitation peuvent limiter la taille du journal des éléments en attente SPX, de sorte que le journal des éléments en attente SPX effectif peut être plus petit que celui demandé ici.

Si le journal des éléments en attente atteint les valeurs indiquées dans le [Tableau 14](#), à la [page 162](#), la connexion SPX est rejetée et le canal ne peut pas démarrer. Pour les canaux de transmission de messages, le canal passe à l'état RETRY et relance la connexion ultérieurement. Pour les connexions client, le client reçoit un code anomalie MQRC_Q_MGR_NOT_AVAILABLE de MQCONN et doit relancer la connexion ultérieurement.

Multi Strophe SSL du fichier qm.ini

La section SSL permet de configurer les canaux TLS sur un gestionnaire de files d'attente.

Protocole OCSP (Online Certificate Status Protocol)

Un certificat peut contenir une extension d'accès AuthorityInfo. Cette extension spécifie un serveur à contacter via le protocole OCSP (Online Certificate Status Protocol). Pour permettre aux canaux SSL ou TLS de votre gestionnaire de files d'attente d'utiliser les extensions d'accès AuthorityInfo, vérifiez que le serveur OCSP qui y est nommé est disponible, qu'il est configuré correctement et qu'il est accessible sur le réseau. Pour plus d'informations, voir [Utilisation des certificats révoqués](#).

CrlDistributionPoint (CDP)

Un certificat peut contenir une extension de point CrlDistribution. Cette extension contient une URL qui identifie à la fois le protocole utilisé pour télécharger une liste de révocation de certificat (CRL) et le serveur à contacter.

Si vous souhaitez autoriser les canaux SSL ou TLS de votre gestionnaire de files d'attente à utiliser les extensions de point CrlDistribution, vérifiez que le serveur CDP qui y est nommé est disponible, correctement configuré et accessible sur le réseau.

La strophe SSL

Utilisez la strophe SSL du fichier `qm.ini` pour configurer la manière dont les canaux TLS de votre gestionnaire de files d'attente tentent d'utiliser les fonctions suivantes et la manière dont ils réagissent en cas de problèmes lors de leur utilisation.

Dans chacun des cas suivants, si la valeur fournie n'est pas l'une des valeurs valides répertoriées, la valeur par défaut est utilisée. Aucun message d'erreur n'est consigné indiquant qu'une valeur non valide a été spécifiée.

OutboundSNI = CHANNEL | NOM_HÔTE

Si **OutboundSNI** est défini sur CHANNEL, les clients compatibles SNI définissent SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS.

Si cet attribut est défini sur HOSTNAME, les clients compatibles SNI définissent l'en-tête SNI sur le nom d'hôte, ce qui entraîne la réception par les demandes de connexion sortantes du certificat par défaut du gestionnaire de files d'attente éloignées lors de l'établissement de liaison TLS. Par conséquent, les certificats par canal ne peuvent pas être utilisés.

Remarque : Si **OutboundSNI=HOSTNAME** est utilisé pour la connexion à un canal distant avec un libellé de certificat configuré, la connexion est rejetée avec un message MQRC_SSL_INITIALIZATION_ERROR et un message [AMQ9673](#) est imprimé dans les journaux d'erreurs du gestionnaire de files d'attente éloignées.

AllowOutboundSNI = YES (valeur par défaut) | NON

Si cette option est activée, les clients compatibles SNI définiront SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS. Si cet attribut est défini sur

NO, les clients compatibles SNI ne définissent pas l'en-tête SNI, ce qui entraîne la réception par les demandes de connexion sortantes du certificat par défaut du gestionnaire de files d'attente éloignées lors de l'établissement de liaison TLS. Par conséquent, les certificats par canal ne peuvent pas être utilisés.



Avertissement : Deprecated Depuis la IBM MQ 9.3.0 , la propriété **AllowOutboundSNI** est obsolète et n'est disponible qu'à des fins de compatibilité avec les versions antérieures.

AllowOutboundSNI défini sur YES fournit la même fonction que **OutboundSNI** défini sur CHANNEL, tandis que **AllowOutboundSNI** défini sur NO fournit la même fonction que **OutboundSNI** défini sur HOSTNAME.

Si les attributs **AllowOutboundSNI** et **OutboundSNI** sont présents dans la strophe SSL, le paramètre **OutboundSNI** est prioritaire.

AllowedCipherSpecs =nom|liste de noms| ALL

Indique une liste personnalisée de CipherSpecs qui sont commandés et activés pour être utilisés avec les canaux IBM MQ sur Multiplatforms.

- Nom CipherSpec unique.
- Liste de noms IBM MQ CipherSpec séparés par des virgules à réactiver.
- Valeur spéciale de ALL, représentant tous les CipherSpecs (non recommandé).

Remarque : Vous ne devez pas sélectionner **ALL** CipherSpecs, car cela active les protocoles SSL 3.0 et TLS 1.0 et un grand nombre d'algorithmes de cryptographie faibles.

Pour plus d'informations, voir [Ajout d'une liste personnalisée de CipherSpecs commandés et activés sur IBM MQ for Multiplatforms](#) dans [CommandeCipherSpec](#) lors de l'établissement de liaison TLS.

IBM i **ALW** **AllowTLSV13 =Y | YES | T | TRUE (valeur par défaut) | N | NO | F | FALSE**

Indique si un gestionnaire de files d'attente peut utiliser les CipherSpecs TLS 1.3 .

- Y (valeur par défaut), YES (valeur par défaut), T (valeur par défaut) ou TRUE (valeur par défaut): Active TLS 1.3 qui permet au gestionnaire de files d'attente d'utiliser les CipherSpecs TLS 1.3 CipherSpecs.
- N, NO, Fou FALSE: désactive TLS 1.3, ce qui signifie que le gestionnaire de files d'attente ne peut pas utiliser TLS 1.3 CipherSpecs.

Pour plus d'informations, voir [Activation des CipherSpecs](#).

CDPCheckExtensions= YES |NO (valeur par défaut)

Indique si les canaux TLS de ce gestionnaire de files d'attente tentent de vérifier les serveurs CDP nommés dans les extensions de certificat de point CrI Distribution.

- YES: Les canaux TLS tentent de vérifier les serveurs CDP pour déterminer si un certificat numérique est révoqué.
- NO (valeur par défaut): les canaux TLS ne tentent pas de vérifier les serveurs CDP. Cette valeur est la valeur par défaut.

V 9.4.0 **HTTPSKeyStore= chaîne**

La chaîne fournit le chemin d'accès à un référentiel de clés pkcs12 qui peut être utilisé par le gestionnaire de files d'attente en tant que magasin de clés de confiance lors de la création de connexions HTTPS sortantes, par exemple à un noeud final JWKS.

Ce fichier doit être chiffré et accompagné d'un fichier 'stash' du même nom, c'est-à-dire d'un fichier avec une extension .sth , utilisé lorsque le gestionnaire de files d'attente doit accéder à ce fichier. Par défaut, si cet attribut n'est pas spécifié, le gestionnaire de files d'attente recherche dans le sous-répertoire ssl des données du gestionnaire de files d'attente un fichier nommé mqdfcer.p12.

Si cet attribut est mis à jour, vous devez redémarrer le gestionnaire de files d'attente pour commencer à utiliser le nouveau fichier de référentiel de clés spécifié.

Voir [Création d'un référentiel de clés à utiliser en tant que magasin de clés de confiance TLS](#) pour savoir comment créer un magasin de clés de confiance basé sur votre environnement de système d'exploitation.

ALW **MinimumRSAKeySize=***int*

Indique la taille de clé minimale que les certificats RSA doivent avoir pour être acceptés lors de l'établissement d'une liaison TLS. Autorise toute valeur égale ou supérieure à 0. La valeur par défaut est 1 si elle n'est pas spécifiée.

OCSPAuthentication=REQUIRED (par défaut) | WARN | FACULTATIF

Indique l'action à effectuer lorsqu'un statut de révocation ne peut pas être déterminé à partir d'un serveur OCSP.

Si la vérification OCSP est activée, un programme de canal TLS tente de contacter un serveur OCSP.

Si le programme de canal ne parvient pas à contacter les serveurs OCSP ou si aucun serveur ne peut fournir le statut de révocation du certificat, la valeur du paramètre OCSPAuthentication est utilisée.

- **REQUIRED** (valeur par défaut): L'échec de la détermination du statut de révocation provoque la fermeture de la connexion avec une erreur. Cette valeur est la valeur par défaut.
- **WARN**: L'échec de la détermination du statut de révocation entraîne l'écriture d'un message d'avertissement dans le journal des erreurs du gestionnaire de files d'attente, mais la connexion est autorisée à se poursuivre.
- **OPTIONAL**: L'échec de la détermination du statut de révocation permet à la connexion de se poursuivre en mode silencieux. Aucun avertissement ou erreur n'est indiqué.

OCSPCheckExtensions = YES (valeur par défaut) | NON

Indique si les canaux TLS de ce gestionnaire de files d'attente tentent de vérifier les serveurs OCSP nommés dans les extensions de certificat d'accès AuthorityInfo.

- **YES** (valeur par défaut): les canaux TLS tentent de vérifier les serveurs OCSP pour déterminer si un certificat numérique est révoqué. Cette valeur est la valeur par défaut.
- **NON**: Les canaux TLS ne tentent pas de vérifier les serveurs OCSP.

ALW **OCSPTimeout=***nombre*

Nombre de secondes d'attente d'un répondeur OCSP lors de l'exécution d'une vérification de révocation.

Depuis la IBM MQ 9.3.0, si la valeur 0 est définie, le délai d'attente par défaut de 30 secondes est utilisé.

Si aucune valeur n'est définie, la valeur par défaut IBM MQ de 30 secondes est utilisée.

SSLHTTPProxyName=*chaîne*

La chaîne correspond au nom d'hôte ou à l'adresse réseau du serveur proxy HTTP qui doit être utilisé par IBM Global Security Kit (GSKit) pour les vérifications OCSP. Cette adresse peut être suivie d'un numéro de port facultatif, entre parenthèses. Si vous n'indiquez pas le numéro de port, le port HTTP par défaut, 80, est utilisé.

AIX Pour les clients 32 bits sous AIX, l'adresse réseau ne peut être qu'une adresse IPv4 .
Sur les autres plateformes, l'adresse réseau peut être une adresse IPv4 ou IPv6 .

Cet attribut peut être nécessaire si, par exemple, un pare-feu empêche l'accès à l'URL du répondeur OCSP.

ALW **PeerCertChainValidation=***chaîne*

La chaîne peut être l'une des deux valeurs suivantes:

- **Usepeerchain [Valeur par défaut]**: La chaîne de certificats fournie par l'homologue peut être utilisée pour combler les lacunes de la chaîne de confiance lors de la validation des certificats. A l'exception du certificat racine.

- **Truststoreonly [Non recommandé]**: seuls les certificats du magasin de clés de confiance seront utilisés pour valider le certificat de l'homologue.

ALW SSLHTTPConnectTimeout= *nombre*|0

Nombre de secondes d'attente pour l'établissement d'une connexion réseau à un serveur HTTP lors de l'exécution d'une vérification de révocation.

Si aucune valeur n'est définie, la valeur par défaut IBM MQ 0 (off) est utilisée.

Exemple de section

```
SSL:
  OutboundSNI=CHANNEL
  AllowedCipherSpecs=TLS13 CipherSpec list
  AllowTLSV13=Y
  CDPCheckExtensions=NO
  MinimumRSAKeySize=1
  OCSPAuthentication=REQUIRED
  OCSPCheckExtensions=YES
  OCSPTimeout=30
  PeerCertChainValidation=Usepeerchain
  SSLHTTPConnectTimeout=0
```

Remarques :

- La valeur par défaut de **OutboundSNI** est **Channel1**.
- La liste **TLS13 CipherSpec** est une liste de CipherSpecs spécifiques et non de chiffrements d'alias. Si vous n'avez besoin que de chiffrements TLS1.3, vous devez les répertorier. Exemple :

```
TLS_CHACHA20_POLY1305_SHA256
TLS_AES_256_GCM_SHA384
TLS_AES_128_GCM_SHA256
TLS_AES_128_CCM_SHA256
TLS_AES_128_CCM_8_SHA256
```

- La valeur par défaut de **AllowTLSV13** est Y sauf si vous avez activé les chiffrements faibles, auquel cas elle est désactivée (sauf si vous l'activez explicitement).
- Les valeurs de **CDPCheckExtensions** peuvent uniquement être Oui ou Non.
- Les valeurs de **PeerCertChainValidation** ne peuvent être que Usepeerchain ou Truststoreonly.

Multi Section de sous-pool du fichier qm.ini

Cette section est créée par IBM MQ. Ne la modifiez pas.

La strophe Subpool et l'attribut **ShortSubpoolName** de cette strophe sont écrits automatiquement par IBM MQ lorsque vous créez un gestionnaire de files d'attente. IBM MQ choisit une valeur pour **ShortSubpoolName**. Ne modifiez pas cette valeur.

Le nom correspond à un répertoire et à un lien symbolique créés dans le répertoire `/var/mqm/sockets`, que IBM MQ utilise pour les communications internes entre ses processus en cours d'exécution.

Multi Section TCP du fichier qm.ini

La strophe TCP spécifie les paramètres de configuration du protocole TCP/IP (Internet Protocol). Ces paramètres remplacent les attributs par défaut des canaux.

Utilisez la section TCP du fichier `qm.ini` pour spécifier les paramètres de configuration TCP/IP.

Windows

Linux

Vous pouvez également, sous Linux (x86 et x86-64) et Windows, utiliser la page des propriétés du gestionnaire de files d'attente TCP IBM MQ Explorer SPX.

Port = 1414 (valeur par défaut) | numéro_port

Numéro de port par défaut, en notation décimale, pour les sessions TCP/IP. Le numéro de port *bien connu* pour IBM MQ est 1414.

Windows Library1= DLLName1 (Windows uniquement)

Nom de la DLL des sockets TCP/IP.

La valeur par défaut est WSOCK32.

Multi SecureCommsOnly = NO | N | FALSE | F (valeur par défaut) | TRUE | T | YES | Y

Indiquez si la communication en texte en clair est autorisée ou non.

SecureCommsOnly=NO | N | FALSE | F

La communication en texte en clair est autorisée et un message d'avertissement est généré lorsque le gestionnaire de files d'attente démarre.

SecureCommsOnly=YES | Y | TRUE | T

La communication en texte brut n'est pas autorisée et un message d'information est généré lorsque le gestionnaire de files d'attente démarre.

KeepAlive = NO (par défaut) | OUI

Activez ou désactivez la fonction KeepAlive . KeepAlive=YES permet à TCP/IP de vérifier périodiquement que l'autre extrémité de la connexion est toujours disponible. Si tel n'est pas le cas, le canal est fermé.

ListenerBacklog= nombre

Remplacez le nombre par défaut de demandes en attente pour le programme d'écoute TCP/IP.

Lors de la réception sur TCP/IP, un nombre maximal de demandes de connexion en attente est défini. Il peut s'agir d'un journal des demandes en attente sur le port TCP/IP pour que le programme d'écoute accepte la demande. Les valeurs de journal des éléments en attente du programme d'écoute par défaut sont indiquées dans le [Tableau 15](#), à la page 167.

Tableau 15. Demandes de connexion en attente par défaut (TCP)	
Plateforme	Valeur ListenerBacklog par défaut
Windows ServeurWindows	100
Linux Linux	100
AIX AIX V5.3 ou ultérieure	100

Remarque : Certains systèmes d'exploitation prennent en charge une valeur supérieure à la valeur par défaut affichée. Utilisez cette option pour éviter d'atteindre le nombre maximal de connexions.

A l'inverse, certains systèmes d'exploitation peuvent limiter la taille du journal des éléments en attente TCP, de sorte que le journal des éléments en attente TCP effectif peut être plus petit que ce qui est demandé ici.

Si le journal des éléments en attente atteint les valeurs indiquées dans le [Tableau 15](#), à la page 167, la connexion TCP/IP est rejetée et le canal ne peut pas démarrer. Pour les canaux de transmission de messages, le canal passe à l'état RETRY et relance la connexion ultérieurement. Pour les connexions client, le client reçoit un code anomalie MQRC_Q_MGR_NOT_AVAILABLE de MQCONN et relance la connexion ultérieurement.

Le groupe de propriétés suivant peut être utilisé pour contrôler la taille des mémoires tampon utilisées par TCP/IP. Les valeurs sont transmises directement à la couche TCP/IP du système d'exploitation. L'utilisation de ces propriétés doit être très soignée. Si les valeurs sont définies de manière incorrecte, cela peut avoir un impact négatif sur les performances TCP/IP. Pour plus d'informations sur la façon dont cela affecte les performances, reportez-vous à la documentation TCP/IP de votre environnement. La valeur zéro indique que le système d'exploitation va gérer les tailles de mémoire tampon, par opposition aux tailles de mémoire tampon fixées par IBM MQ.

Connect_Timeout = 0 (valeur par défaut) | nombre

Nombre de secondes avant qu'une tentative de connexion du socket n'expire. La valeur par défaut zéro indique qu'il n'y a pas de délai de connexion.

Les processus de canal IBM MQ se connectent via des sockets non bloquants. Par conséquent, si l'autre extrémité du socket n'est pas prête, connect () renvoie immédiatement *EINPROGRESS* ou *EWOULDBLOCK*. Par la suite, la connexion sera tentée à nouveau, jusqu'à un total de 20 tentatives de ce type, lorsqu'une erreur de communication est signalée.

Si Connect_Timeout est défini sur une valeur différente de zéro, IBM MQ attend la période stipulée sur l'appel select () pour que le socket soit prêt. Cela augmente les chances de réussite d'un appel connect () ultérieur. Cette option peut être utile dans les cas où les connexions nécessitent un délai d'attente, en raison d'une charge élevée sur le réseau.

SndBuffSize = nombre | 0 (valeur par défaut)

Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité émettrice des canaux. Cette valeur de section peut être remplacée par une section plus spécifique au type de canal, par exemple RcvSndBuffSize. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ , 32768, est utilisée.

Multi Les nouveaux gestionnaires de files d'attente sont automatiquement créés avec la valeur par défaut 0 (voir «Exemple de section», à la page 169).

RcvBuffSize = nombre | 0 (valeur par défaut)

Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité réceptrice des canaux. Cette valeur de section peut être remplacée par une section plus spécifique au type de canal, par exemple RcvRcvBuffSize. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ , 32768, est utilisée.

Multi Les nouveaux gestionnaires de files d'attente sont automatiquement créés avec la valeur par défaut 0 (voir «Exemple de section», à la page 169).

RcvSndBuffSize = nombre | 0 (valeur par défaut)

Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité émettrice d'un canal récepteur. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ , 32768, est utilisée.

Multi Les nouveaux gestionnaires de files d'attente sont automatiquement créés avec la valeur par défaut 0 (voir «Exemple de section», à la page 169).

RcvRcvBuffSize = nombre | 0 (valeur par défaut)

Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité réceptrice d'un canal récepteur. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ , 32768, est utilisée.

Multi Les nouveaux gestionnaires de files d'attente sont automatiquement créés avec la valeur par défaut 0 (voir «Exemple de section», à la page 169).

SvrSndBuffSize = nombre | 0 (valeur par défaut)

Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité serveur d'un canal de connexion serveur de connexion client. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ , 32768, est utilisée.

Multi Les nouveaux gestionnaires de files d'attente sont automatiquement créés avec la valeur par défaut 0 (voir «Exemple de section», à la page 169).

SvrRcvBuffSize = nombre | 0 (valeur par défaut)

Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité serveur d'un canal de connexion serveur de connexion client. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ , 32768, est utilisée.

Multi Les nouveaux gestionnaires de files d'attente sont automatiquement créés avec la valeur par défaut 0 (voir «Exemple de section», à la page 169).

Exemple de section

```
TCP:
  SndBuffSize=0
  RcvBuffSize=0
  RcvSndBuffSize=0
  RcvRcvBuffSize=0
  ClntSndBuffSize=0
  ClntRcvBuffSize=0
  SvrSndBuffSize=0
  SvrRcvBuffSize=0
```

Remarque : **Multi** Pour les nouveaux gestionnaires de files d'attente sur Multiplatforms, les tailles de mémoire tampon d'envoi et de réception TCP par défaut dans la strophe TCP de `qm.ini` file sont définies pour être gérées par le système d'exploitation. Comme illustré dans l'exemple précédent, les nouveaux gestionnaires de files d'attente sont créés automatiquement avec la valeur par défaut 0 pour les mémoires tampon d'envoi et de réception. S'applique uniquement aux nouveaux gestionnaires de files d'attente. Les paramètres de mémoire tampon d'envoi et de réception TCP pour les gestionnaires de files d'attente migrés à partir de versions antérieures d' IBM MQ sont conservés.

Si les propriétés de taille de la mémoire tampon TCP sont supprimées du fichier `qm.ini`, la mémoire tampon par défaut est définie sur 32K. Vous devez être prudent lorsque vous utilisez cette valeur par défaut car 32K peut ne pas être une mémoire tampon appropriée pour tous les scénarios de messagerie.

Si les propriétés de mémoire tampon d'envoi et de réception TCP sont définies sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. La méthode de sélection de ces valeurs par défaut varie en fonction du système d'exploitation, mais peut généralement être trouvée dans les pages de manuel du système d'exploitation "tcp" ou `get/setsockopt ()`.

Multi Section TuningParameters du fichier qm.ini

La section TuningParameters spécifie les options d'optimisation du gestionnaire de files d'attente.

SuppressDspAuthFail= YES |NO (valeur par défaut)

Lorsque la valeur est YES, le gestionnaire de files d'attente supprime la génération des événements d'autorisation et l'écriture des messages d'erreur AMQ8077 dans le journal des erreurs en cas d'échec d'un contrôle d'autorisation, si la connexion ne dispose pas des droits + dsp sur un objet.

ImplSyncOpenOutput=valeur

ImplSyncOpenOutput est le nombre minimal d'applications pour lesquelles la file d'attente est ouverte pour insertion, avant qu'un point de synchronisation implicite puisse être activé pour une insertion permanente, en dehors du point de synchronisation. La valeur par défaut de **ImplSyncOpenOutput** est 2.

Cela a pour effet que si une seule application a cette file d'attente ouverte pour une opération d'insertion, **ImplSyncOpenOutput** est désactivé.

La spécification de `ImplSyncOpenOutput=1` signifie qu'un point de synchronisation implicite est toujours pris en compte. Vous pouvez définir n'importe quelle valeur entière positive.

Si vous ne souhaitez jamais ajouter de point de synchronisation implicite, définissez `ImplSyncOpenOutput=OFF`.

UniformClusterNom =nom du cluster

Nom du cluster IBM MQ que vous utilisez comme cluster uniforme.

OAMLdapConnectTimeout=heure|0 (valeur par défaut)

Durée maximale, en secondes, pendant laquelle le client LDAP attend pour établir une connexion TCP au serveur. Si vous fournissez plusieurs serveurs LDAP via une liste de noms de connexion, le

décalé d'attente s'applique à chaque tentative de connexion individuelle, de sorte qu'une connexion à l'entrée suivante de la liste de noms est tentée si ce délai d'attente est atteint.

`time` a une valeur maximale de 3600 secondes et une valeur de 0, qui correspond à la valeur minimale et à la valeur par défaut, signifie que l'attente est illimitée.

OAMLdapQueryTimeLimit=time|0 (valeur par défaut)

Durée maximale, en secondes, pendant laquelle le client LDAP attend une réponse à une demande LDAP du serveur, une fois qu'une connexion a été établie et qu'une demande LDAP a été envoyée.

`time` a une valeur maximale de 3600 secondes et une valeur de 0, qui correspond à la valeur minimale et à la valeur par défaut, signifie que l'attente est illimitée.

OAMLdapResponseWarningTime=seuil

Si une connexion à un serveur LDAP a pris plus de temps que le nombre de secondes défini par le paramètre **OAMLdapResponseWarningTime**, un message [AMQ5544W](#) est consigné dans le journal des erreurs. Le seuil par défaut est de 10 secondes.

ExpiryInterval

Indique la fréquence à laquelle le gestionnaire de files d'attente analyse les files d'attente à la recherche de messages arrivés à expiration qui n'ont pas encore été nettoyés par d'autres activités de file d'attente. Il s'agit d'un intervalle de temps en secondes.

Par défaut, le scanner d'expiration s'exécute environ toutes les cinq minutes sur les générations IBM MQ de production.



ATTENTION : La modification de la valeur **ExpiryInterval** n'est généralement pas requise et vous devez la modifier uniquement sous la supervision du support IBM .

LivenessHeartBeatLen

Configure la fréquence à laquelle les vérifications du gestionnaire de files d'attente qui écrivent dans le journal sont effectuées à une fréquence raisonnable. La valeur maximale pour **LivenessHeartBeatLen** est de 600 secondes (10 minutes) et la valeur minimale est de 0, ce qui a pour effet de désactiver complètement la vérification.



ATTENTION : Dans la plupart des cas, il n'est pas nécessaire de modifier la fréquence de ces contrôles. N'effectuez aucune modification sauf si le support IBM vous le recommande.

ECHeartBeatLen

Configure la fréquence des diagnostics d'intégrité généraux du gestionnaire de files d'attente. La valeur minimale pour **ECHeartBeatLen** est 10000 millisecondes (10 secondes) et la valeur maximale est 60000 millisecondes (60 secondes).



ATTENTION : Dans la plupart des cas, il n'est pas nécessaire de modifier la fréquence de ces contrôles. N'effectuez aucune modification sauf si le support IBM vous le recommande.

FileLockHeartBeatLen

Modifie la valeur par défaut du verrou de fichier pour un gestionnaire de files d'attente multi-instance que le contrôleur d'exécution exécute régulièrement afin de s'assurer qu'il détient toujours le verrou exclusif sur le fichier multi-instance principal. Par défaut, ces vérifications de verrouillage de fichier sont effectuées toutes les 20 secondes. La valeur minimale pour **FileLockHeartBeatLen** est de 10 secondes et la valeur maximale est de 600 secondes (10 minutes).



ATTENTION : Dans la plupart des cas, il n'est pas nécessaire de modifier la fréquence de ces contrôles. N'effectuez aucune modification sauf si le support IBM vous le recommande.

Exemple de section

```
TuningParameters:
  SuppressDspAuthFail=NO
  ImplSyncOpenOutput=2
  OAMLdapConnectTimeout=60
  OAMLdapQueryTimeLimit=60
  OAMLdapResponseWarningTime=10
  ExpiryInterval=300
```

Concepts associés

[Point de synchronisation implicite](#)

Multi **Strophe Variables du fichier qm.ini**

La section Variables spécifie les variables de configuration à utiliser avec les clusters uniformes automatiques.

Vous pouvez utiliser les attributs répertoriés dans la section Variables lors de la configuration de cluster automatique de CONNAME et les zones MQSC de nom de canal d'un canal récepteur de cluster. Les variables de configuration ne peuvent pas être utilisées dans un autre élément d'un script MQSC.

attribute=valeur

Indique un nom et une valeur associée à utiliser comme insertion lors des définitions MQSC.

Les paires *attribute=value* peuvent être fournies à l'aide de l'option de ligne de commande **-iv** de la commande **crtmqm** lorsque vous créez un gestionnaire de files d'attente.

Exemple de section

```
Variables:  
  CONNAME=127.0.0.1(1414)
```

Concepts associés

[«Équilibrage automatique des applications», à la page 447](#)

L'équilibrage automatique des applications améliore considérablement la distribution et la disponibilité des applications en permettant à un cluster uniforme IBM MQ de gérer étroitement la distribution des applications dans le cluster, plutôt que de s'appuyer sur la randomisation ou sur une épingle manuel des applications à des gestionnaires de files d'attente spécifiques.

Tâches associées

[«Création d'un nouveau cluster uniforme», à la page 461](#)

Comment créer un nouveau cluster uniforme.

Référence associée

[«Utilisation de la configuration de cluster automatique», à la page 465](#)

Vous configurez IBM MQ pour activer la configuration automatique en modifiant les informations de configuration `qm.ini`.

Multi **Section XAResourceManager du fichier qm.ini**

La section XAResourceManager permet de spécifier des informations sur les gestionnaires de ressources impliqués dans les unités d'oeuvre globales coordonnées par le gestionnaire de files d'attente.

Utilisez la strophe XAResourceManager du fichier `qm.ini` pour spécifier les informations sur les gestionnaires de ressources impliqués dans les unités d'oeuvre globales coordonnées par le gestionnaire de files d'attente.

Windows **Linux** Sinon, sous Linux (x86 et x86-64) et Windows, utilisez la IBM MQ Explorer page de propriétés du gestionnaire de files d'attente du gestionnaire de ressources XA.

Ajoutez manuellement des informations de configuration de gestionnaire de ressources XA pour chaque instance d'un gestionnaire de ressources participant à des unités d'oeuvre globales ; aucune valeur par défaut n'est fournie.

Pour plus d'informations sur les attributs du gestionnaire de ressources, voir [Coordination de la base de données](#).

Nom = nom (obligatoire)

Cet attribut identifie l'instance du gestionnaire de ressources.

La valeur Name peut comporter jusqu'à 31 caractères. Vous pouvez utiliser le nom du gestionnaire de ressources tel qu'il est défini dans sa structure de commutateur XA. Toutefois, si vous utilisez plusieurs instances du même gestionnaire de ressources, vous devez construire un nom unique pour chaque instance. Vous pouvez garantir l'unicité en incluant le nom de la base de données dans la chaîne Name , par exemple.

IBM MQ utilise la valeur Name dans les messages et dans la sortie de la commande dspmqtrn .

Ne modifiez pas le nom d'une instance de gestionnaire de ressources et ne supprimez pas son entrée des informations de configuration une fois que le gestionnaire de files d'attente associé a démarré et que le nom du gestionnaire de ressources est actif.

SwitchFile= nom (obligatoire)

Nom qualifié complet du fichier de chargement contenant la structure de commutateur XA du gestionnaire de ressources.

Si vous utilisez un gestionnaire de files d'attente 64 bits avec des applications 32 bits, la valeur name doit contenir uniquement le nom de base du fichier de chargement contenant la structure de commutation XA du gestionnaire de ressources.

Le fichier 32 bits sera chargé dans l'application à partir du chemin spécifié par ExitsDefaultPath.

Le fichier 64 bits sera chargé dans le gestionnaire de files d'attente à partir du chemin indiqué par ExitsDefaultPath64.

XAOpenString= chaîne (facultatif)

Chaîne de données à transmettre au point d'entrée xa_open du gestionnaire de ressources. Le contenu de la chaîne dépend du gestionnaire de ressources lui-même. Par exemple, la chaîne peut identifier la base de données à laquelle cette instance du gestionnaire de ressources doit accéder. Pour plus d'informations sur la définition de cet attribut, voir:

- [Ajout des informations de configuration du gestionnaire de ressources pour Db2](#)
- [Ajout d'informations de configuration de gestionnaire de ressources pour Oracle](#)
- [Ajout d'informations de configuration de gestionnaire de ressources pour Sybase](#)
- [Ajout des informations de configuration du gestionnaire de ressources pour Informix](#)

et consultez la documentation de votre gestionnaire de ressources pour connaître la chaîne appropriée.


XACloseString= chaîne (facultatif)

Chaîne de données à transmettre au point d'entrée xa_close du gestionnaire de ressources. Le contenu de la chaîne dépend du gestionnaire de ressources lui-même. Pour plus d'informations sur la définition de cet attribut, voir:

- [Ajout des informations de configuration du gestionnaire de ressources pour Db2](#)
- [Ajout d'informations de configuration de gestionnaire de ressources pour Oracle](#)
- [Ajout d'informations de configuration de gestionnaire de ressources pour Sybase](#)
- [Ajout des informations de configuration du gestionnaire de ressources pour Informix](#)

et consultez la documentation de votre base de données pour connaître la chaîne appropriée.

ThreadOfControl=THREAD | PROCESS

 Cet attribut est obligatoire pour Windows. Le gestionnaire de files d'attente utilise cette valeur pour la sérialisation lorsqu'il doit appeler le gestionnaire de ressources à partir de l'un de ses propres processus à unités d'exécution multiples.

UNITE D'EXECUTION

Le gestionnaire de ressources est entièrement *sensible aux unités d'exécution*. Dans un processus IBM MQ à unités d'exécution multiples, des appels de fonction XA peuvent être émis simultanément vers le gestionnaire de ressources externe à partir de plusieurs unités d'exécution.

PROCESS

Le gestionnaire de ressources n'est pas *autorisant les unités d'exécution multiples*. Dans un processus IBM MQ à unités d'exécution multiples, un seul appel de fonction XA à la fois peut être effectué vers le gestionnaire de ressources.

L'entrée **ThreadOfControl** ne s'applique pas aux appels de fonction XA émis par le gestionnaire de files d'attente dans un processus d'application à unités d'exécution multiples. En général, une application comportant des unités d'oeuvre simultanées sur différentes unités d'exécution requiert que ce mode de fonctionnement soit pris en charge par chacun des gestionnaires de ressources.

Exemple de section

```
XAResourceManager:
  Name=DB2 Resource Manager Bank
  SwitchFile=/usr/bin/db2swit
  XAOpenString=MQBankDB
  XACloseString=
  ThreadOfControl=THREAD
```

Remarque : Le nombre maximal de strophes XAResourceManager est limité à 255. Toutefois, vous ne devez utiliser qu'un petit nombre de strophes pour éviter une dégradation des performances des transactions.

IBM i

Exemple de fichier qm.ini pour IBM i

Exemple montrant comment des groupes d'attributs peuvent être organisés dans un fichier de configuration de gestionnaire de files d'attente pour IBM i.

```
#####
#* Module Name: qm.ini                               *#
#* Type       : IBM MQ queue manager configuration file *#
#* Function   : Define the configuration of a single queue manager *#
#*          *#
#####
#* Notes      : *#
#* 1) This file defines the configuration of the queue manager *#
#*          *#
#####
Log:
LogPath=QMSATURN.Q
LogReceiverSize=65536

CHANNELS:
MaxChannels = 20      ; Maximum number of channels allowed.
                    ; Default is 100.
MaxActiveChannels = 10 ; Maximum number of channels allowed to be
                    ; active at any time. The default is the
                    ; value of MaxChannels.

TCP:
KeepAlive = Yes      ; TCP/IP entries.
                    ; Switch KeepAlive on.
SvrSndBuffSize=20000 ; Size in bytes of the TCP/IP send buffer for each
                    ; channel instance. Default is 32768.
SvrRcvBuffSize=20000 ; Size in bytes of the TCP/IP receive buffer for each
                    ; channel instance. Default is 32768.
Connect_Timeout=10000 ; Number of seconds before an attempt to connect the
                    ; channel instance times out. Default is zero (no timeout).

QMErrorLog:
ErrorLogSize = 262144
ExcludeMessage = 7234
SuppressMessage = 9001,9002,9202
SuppressInterval = 30

TuningParameters:
ImplSyncOpenOutput=2
```

Fichier de configuration de l'installation, mqinst.ini

Sur les systèmes AIX and Linux , le fichier de configuration d'installation, `mqinst.ini`, contient des informations sur toutes les installations IBM MQ . Sur les systèmes Windows , les informations de configuration de l'installation se trouvent dans le registre.

Emplacement du fichier mqinst.ini

Linux

AIX

Le fichier `mqinst.ini` se trouve dans le répertoire `/etc/opt/mqm` sur les systèmes AIX and Linux . Il contient des informations sur l'installation principale, le cas échéant, ainsi que les informations suivantes pour chaque installation:

- Nom de l'installation
- Description de l'installation
- Identificateur d'installation
- Chemin d'installation

Important : Le fichier `mqinst.ini` ne doit pas être édité ou référencé directement car son format n'est pas fixe et peut changer.

L'identificateur d'installation, à usage interne uniquement, est défini automatiquement et ne doit pas être modifié.

Au lieu d'éditer directement le fichier `mqinst.ini`, vous devez utiliser les commandes suivantes pour créer, supprimer, interroger et modifier les valeurs du fichier:

`crtmqinst` pour créer des entrées.

`dltmqinst` pour supprimer des entrées.

`dspmqinst` pour afficher les entrées.

`setmqinst` pour définir des entrées.

Informations de configuration de l'installation sous Windows

Windows

Il n'existe pas de fichier `mqinst.ini` sous Windows. Les informations de configuration de l'installation se trouvent dans le registre et se trouvent dans la clé suivante:

```
HKLM\SOFTWARE\IBM\WebSphere MQ\Installation\InstallationName
```

Important : Cette clé ne doit pas être éditée ou référencée directement car son format n'est pas fixe et peut changer.

A la place, vous devez utiliser les commandes suivantes pour interroger et modifier les valeurs du registre:

`dspmqinst` pour afficher les entrées.

`setmqinst` pour définir des entrées.

Sous Windows, les commandes `crtmqinst` et `dltmqinst` ne sont pas disponibles. Les processus d'installation et de désinstallation gèrent la création et la suppression des entrées de registre requises.

Fichier de configuration IBM MQ MQI client , mqclient.ini

Vous pouvez configurer vos clients en utilisant des attributs dans un fichier texte. Ces attributs peuvent être remplacés par des variables d'environnement ou par d'autres méthodes spécifiques à la plateforme.

Vous configurez votre IBM MQ MQI clients à l'aide d'un fichier texte, similaire au fichier de configuration du gestionnaire de files d'attente, `qm.ini`. Le fichier contient un certain nombre de strophes, chacune contenant un certain nombre de lignes au format **attribute-name = valeur**.

Le fichier de configuration IBM MQ MQI client est généralement nommé `mqclient.ini`, mais vous pouvez choisir de lui attribuer un autre nom. Les informations de configuration de ce fichier s'appliquent aux plateformes suivantes:

- **ALW** AIX, Linux, and Windows
- **IBM i** IBM i

Remarque : Sous IBM i, il n'existe pas de fichier `mqclient.ini` par défaut. Toutefois, vous pouvez créer le fichier dans IBM i Integrated File System (IFS).

Pour plus d'informations, voir «[Emplacement du fichier de configuration du client](#)», à la page 176.

Remarque : **z/OS** La plateforme z/OS ne peut pas être utilisée pour exécuter des clients IBM MQ. Par conséquent, le fichier `mqclient.ini` n'existe pas sur IBM MQ for z/OS.

Les attributs du fichier de configuration IBM MQ MQI client s'appliquent aux clients qui utilisent:

- L'interface MQI
- IBM MQ classes for Java
- IBM MQ classes for JMS
- IBM MQ classes for .NET
- XMS

Bien que les attributs du fichier de configuration IBM MQ MQI client s'appliquent à la plupart des clients IBM MQ, certains attributs ne sont pas lus par les clients .NET et XMS .NET gérés ou par les clients qui utilisent le IBM MQ classes for Java ou le IBM MQ classes for JMS. Pour plus d'informations, voir «[Quels sont les clients IBM MQ qui peuvent lire chaque attribut](#)», à la page 177.

Les fonctions de configuration s'appliquent à toutes les connexions établies par une application client à des gestionnaires de files d'attente, au lieu d'être spécifiques à une connexion individuelle à un gestionnaire de files d'attente. Les attributs relatifs à une connexion à un gestionnaire de files d'attente individuel peuvent être configurés à l'aide d'un programme, par exemple à l'aide d'une structure MQCD ou d'une table de définition de canal du client (CCDT).

Voici un exemple de fichier de configuration client:

```
## Module Name: mqclient.ini                ##
## Type       : IBM MQ MQI client configuration file      ##
## Function   : Define the configuration of a client     ##
##           ##
## Notes     :                                         ##
## 1) This file defines the configuration of a client    ##
##           ##
##           ##
ClientExitPath:
  ExitsDefaultPath=/var/mqm/exits
  ExitsDefaultPath64=/var/mqm/exits64

TCP:
  Library1=DLLName1
  KeepAlive = Yes
  ClntSndBuffSize=32768
  ClntRcvBuffSize=32768
  Connect_Timeout=0

MessageBuffer:
  MaximumSize=-1
  Updatepercentage=-1
  PurgeTime=0

LU62:
```

```

TPName
Library1=DLLName1
Library2=DLLName2

PreConnect:
Module=myMod
Function=myFunc
Data=ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
Sequence=1

CHANNELS:
DefRecon=YES
ServerConnectionParms=SALES.SVRCONN/TCP/hostname.x.com(1414)

Connection:
App1Name=ExampleApp1Name

```

Vous ne pouvez pas configurer plusieurs connexions de canal à l'aide du fichier de configuration du client.

Les variables d'environnement qui étaient prises en charge dans les versions antérieures à IBM WebSphere MQ 7.0 continuent d'être prises en charge dans les versions ultérieures, et lorsqu'une telle variable d'environnement correspond à une valeur équivalente dans le fichier de configuration du client, la variable d'environnement remplace la valeur du fichier de configuration du client.

Pour une application client qui utilise IBM MQ classes for JMS, vous pouvez également remplacer le fichier de configuration du client comme suit:

- En définissant des propriétés dans le fichier de configuration JMS .
- En définissant les propriétés système Java , qui remplacent également le fichier de configuration JMS .

Pour le client .NET , vous pouvez également remplacer le fichier de configuration du client et les variables d'environnement équivalentes à l'aide du fichier de configuration de l'application .NET .

Commentaires dans le fichier de configuration

Linux AIX

Vous pouvez utiliser le point-virgule ';' et le hachage '#' pour marquer le début d'un commentaire dans le fichier de configuration. Cela peut marquer une ligne entière en tant que commentaire ou indiquer un commentaire à la fin d'une ligne qui ne sera pas incluse dans la valeur d'un paramètre.

Si une valeur requiert l'un de ces caractères, vous devez le mettre en échappement à l'aide de la barre oblique inversée '\'.

L'exemple suivant illustre l'utilisation des commentaires dans le fichier de configuration:

```

# Example of an SSL stanza with comments
SSL:
    ClientRevocationChecks=REQUIRED ; Example of an end of line comment
    SSLCryptoHardware=GSK_PKCS11=/driver\;label\;password\;SYMMETRIC_CIPHER_ON # Example of
    escaped comment characters.

```

Concepts associés

[Le fichier de configuration IBM MQ classes for Java](#)

Multi

Emplacement du fichier de configuration du client

Un fichier de configuration IBM MQ MQI client peut être stocké dans un certain nombre d'emplacements.

Une application client utilise le chemin de recherche suivant pour localiser le fichier de configuration IBM MQ MQI client :

1. Emplacement spécifié par la variable d'environnement **MQCLNTCF**.



Le format de cette variable d'environnement est une URL complète. Cela signifie que le nom de fichier n'est pas nécessairement `mqclient.ini` et facilite le placement du fichier sur un système de fichiers connecté au réseau.


Remarques :

- Les clients C, .NET et XMS prennent en charge uniquement le protocole file: ; le protocole file: est utilisé si la chaîne d'URL ne commence pas par protocol:
 - Pour autoriser les environnements d'exécution Java Java 1.4.2, qui ne prennent pas en charge la lecture des variables d'environnement, la variable d'environnement **MQCLNTCF** peut être remplacée par une propriété système **MQCLNTCF** Java.
2. Un fichier appelé mqclient.ini dans le répertoire de travail actuel de l'application.
 3. Un fichier appelé mqclient.ini dans le répertoire de données IBM MQ pour les systèmes AIX, Linux, and Windows.

Remarques :


- Le répertoire de données IBM MQ n'existe pas dans les cas suivants:




-  sur IBM i
-  sur z/OS
- lorsque le client a été fourni avec un autre produit

 Sous IBM i, il n'existe pas de fichier mqclient.ini par défaut. Toutefois, le fichier peut être créé dans le système de fichiers IBM i Integrated File System (IFS) dans le répertoire /QIBM/UserData/mqm/, et la variable d'environnement **MQCLNTCF** peut être définie pour pointer vers ce fichier. Exemple :




```
ADDENVVAR ENVVAR(MQCLNTCF) VALUE('QIBM/UserData/mqm/mqclient.ini') REPLACE(*YES)
```

Pour plus d'exemples de variables d'environnement, voir «Description des variables d'environnement», à la page 68.

 La plateforme z/OS ne peut pas être utilisée pour exécuter des clients IBM MQ. Par conséquent, le fichier mqclient.ini n'existe pas sur IBM MQ for z/OS.

-   Sur les systèmes AIX and Linux, le répertoire est /var/mqm.
-  Sur les plateformes Windows, vous configurez la variable d'environnement **MQ_DATA_PATH** lors de l'installation pour qu'elle pointe vers le répertoire de données. Il s'agit normalement de C:\ProgramData\IBM\MQ.

Remarque : Si vous installez un client uniquement, la variable d'environnement peut être **MQ_FILE_PATH**.

- Pour autoriser les environnements d'exécution Java Java 1.4.2 qui ne prennent pas en charge la lecture des variables d'environnement, vous pouvez remplacer manuellement la variable d'environnement **MQ_DATA_PATH** par une propriété système **MQ_DATA_PATH** Java.
4. Un fichier appelé mqclient.ini dans un répertoire standard approprié à la plateforme et accessible aux utilisateurs:
 - Pour tous les clients Java, il s'agit de la valeur de la propriété système user.home Java.
 -   Pour les clients C sur les plateformes AIX and Linux, il s'agit de la valeur de la variable d'environnement **HOME**.
 -  Pour les clients C sous Windows, il s'agit des valeurs concaténées des variables d'environnement **HOMEDRIVE** et **HOMEPATH**.

 **Quels sont les clients IBM MQ qui peuvent lire chaque attribut**

La plupart des attributs du fichier de configuration IBM MQ MQI client peuvent être utilisés par le client C et les clients .NET non gérés. Toutefois, certains attributs ne sont pas lus par les clients .NET et XMS .NET gérés ou par les clients qui utilisent IBM MQ classes for Java ou IBM MQ classes for JMS.

Tableau 16. Les attributs qui s'appliquent à chaque type de client

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
Strophe CHANNELS						
<u>ID de jeu de caractères codés</u>	Numéro de jeu de caractères codés à utiliser.	Oui	Non	Non	Oui	Oui
<u>RépertoireChannelDefinition</u>	Chemin de répertoire du fichier contenant la table de définition de canal du client.	Oui	Non	Non	Oui	Oui
<u>FichierChannelDefinition</u>	Nom du fichier contenant la table de définition de canal du client.	Oui	Non	Non	Oui	Oui
<u>ReconDelay</u>	Option d'administration permettant de configurer le délai de reconnexion pour les programmes client qui peuvent se reconnecter automatiquement.	Oui	Non	Oui	Oui	Oui

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>DefRecon</u>	Option d'administration permettant aux programmes client de se reconnecter automatiquement ou de désactiver la reconnexion automatique d'un programme client qui a été écrit pour se reconnecter automatiquement.	Oui	Non	Oui	Oui	Oui
<u>MQReconnectTimeout</u>	Délai d'attente en secondes pour se reconnecter à un client.	Oui	Non	Non	Oui	Non
<u>ServerConnectionParameters</u>	Emplacement du serveur IBM MQ et méthode de communication à utiliser.	Oui	Non	Non	Oui	Oui
<u>Put1DefaultAlwaysSync</u>	Contrôle le comportement de l'appel de fonction MQPUT1 avec l'option MQPMO_RESPONSE_AS_Q_DEF.	Oui	Oui	Oui	Oui	Oui

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>PasswordProtection</u>	Permet de définir des mots de passe protégés dans la structure MQCSP, plutôt que d'utiliser SSL ou TLS.	Oui	Oui	Oui	Oui	Oui
ClientExitPath						
<u>ExitsDefaultPath</u>	Indique l'emplacement des exits de canal 32 bits pour les clients.	Oui	Oui	Oui	Oui	Oui
<u>ExitsDefaultPath64</u>	Indique l'emplacement des exits de canal 64 bits pour les clients.	Oui	Oui	Oui	Oui	Oui
<u>JavaExitsClassPath</u>	Valeurs à ajouter au chemin d'accès aux classes lors de l'exécution d'un exit Java .	Non	Oui	Oui	Non	Non
Strophe de connexion						
<u>AppName</u>	Nom d'application spécifié dans le fichier de configuration du client.	Oui	Non	Non	Non	Non
Strophe JMQUI						

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>useMQCSPAuthentication</u>	Contrôle si les applications IBM MQ classes for Java et IBM MQ classes for JMS doivent utiliser le mode compatibilité ou le mode d'authentification MQCSP lors de l'authentification avec un gestionnaire de files d'attente.	Non	Oui	Oui	Non	Non
sectionMessageBuffer						
<u>MaximumSize</u>	Taille, en kilooctets, de la mémoire tampon de lecture anticipée, comprise entre 1 et 999 999.	Oui	Oui	Oui	Oui	Oui
<u>PurgeTime</u>	Intervalle, en secondes, après lequel les messages laissés dans la mémoire tampon de lecture anticipée sont purgés.	Oui	Oui	Oui	Oui	Oui

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>UpdatePercentage</u>	Valeur de pourcentage de mise à jour, comprise entre 1 et 100, utilisée pour calculer la valeur de seuil afin de déterminer à quel moment une application client effectue une nouvelle demande auprès du serveur.	Oui	Oui	Oui	Oui	Oui
sectionPreConnect						
<u>Données</u>	URL du référentiel dans lequel les définitions de connexion sont stockées.	Oui	Non	Non	Non	Non
<u>Fonction</u>	Nom du point d'entrée fonctionnel dans la bibliothèque qui contient le code d'exit PreConnect .	Oui	Non	Non	Non	Non
<u>Module</u>	Nom du module contenant le code d'exit d'API.	Oui	Non	Non	Non	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>Sequence</u>	Séquence dans laquelle cet exit est appelé par rapport à d'autres exits.	Oui	Non	Non	Non	Non
Strophe de sécurité						
<u>DisableClientAMS</u>	Désactive ou active AMS pour les connexions client à un gestionnaire de files d'attente.	Oui	Oui	Oui	Non	Non
Strophe SSL						
<u>OutboundSNI</u>	Indique si les clients compatibles SNI vont définir SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS ou sur le nom d'hôte.	Oui	Oui	Oui	Oui	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)



Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>AllowOutboundSNI</u>	<p>Indique si les clients compatibles SNI définiront SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS.</p> <p> Avertissement :  Depuis la IBM MQ 9.3.0 , cette propriété est obsolète. Utilisez OutboundSNI à la place.</p>	Oui	Oui	Oui	Non	Non
<u>AllowTLV13</u>	Indique si un gestionnaire de files d'attente peut utiliser les CipherSpecs TLS 1.3 .	Oui (clients C/C++)	Non	Non	Non	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
CDPCheckExtensions	Indique si les canaux SSL ou TLS de ce gestionnaire de files d'attente tentent de vérifier les serveurs CDP nommés dans les extensions de certificat de point CrLDistribution.	Oui	Non	Non	Non	Non
CertificateLabel	Libellé de certificat de la définition de canal.	Oui	Non	Non	Non	Non
PolitiqueCertificateValidation	Détermine le type de validation de certificat utilisé.	Oui	Non	Non	Non	Non
ClientRevocationVerifications	Détermine comment la vérification de la révocation de certificat est configurée si l'appel de connexion client utilise un canal SSL/TLS.	Oui	Non	Non	Non	Non
EncryptionPolicySuiteB	Détermine si un canal utilise la cryptographie compatible Suite-B et le niveau de puissance à utiliser.	Oui	Non	Non	Non	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>Environment Scope</u>	Contrôle si IBM MQ utilise un environnement IBM Global Security Kit (GSKit) unique pour l'ensemble du processus ou un environnement GSKit par connexion.	Oui (clients C)	Non	Non	Non	Non
<u>MinimumRSA KeyTaille</u>	Indique la taille de clé minimale que les certificats RSA doivent avoir pour être acceptés.	Oui (clients C/C++)	Non	Non	Non	Non
<u>Authentication OCSPAuthentication</u>	Définit le comportement de IBM MQ lorsque OCSP est activé et que la vérification de révocation OCSP ne parvient pas à déterminer le statut de révocation de certificat.	Oui	Non	Non	Non	Non
<u>OCSPCheckExtensions</u>	Contrôle si IBM MQ agit sur les extensions de certificat d'accès AuthorityInfo.	Oui	Non	Non	Non	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)


Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>Délai d'attente OCSPTimeout</u>	Nombre de secondes d'attente d'un répondeur OCSP lors de l'exécution d'une vérification de révocation.	Oui	Non	Non	Non	Non
 <u>PeerCertChainValidation</u>	Paramètre de validation de certificat GSKit .	Oui	Non	Non	Non	Non
<u>SSLCryptoHardware</u>	Définit la chaîne de paramètres requise pour configurer le matériel de cryptographie PKCS #11 présent sur le système.	Oui	Non	Non	Non	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
SSLCryptoHardwareKeyFile	Indique le chemin d'accès complet et le nom du fichier contenant la clé initiale qui a été utilisée pour chiffrer le mot de passe dans la chaîne de configuration du matériel de cryptographie PKCS #11 spécifiée avec l'attribut SSLCryptoHardware .	Oui	Non	Non	Non	Non
SSLFipsRequired	Indique si seuls les algorithmes certifiés FIPS doivent être utilisés si la cryptographie est effectuée dans IBM MQ.	Oui	Non	Non	Non	Non
SSLHTTPProxyName	La chaîne correspond au nom d'hôte ou à l'adresse réseau du serveur proxy HTTP qui doit être utilisé par GSKit pour les vérifications OCSP.	Oui	Non	Non	Non	Non

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
SSLHTTPConnectTimeout	Nombre de secondes d'attente pour l'établissement d'une connexion réseau à un serveur HTTP lors de l'exécution d'une vérification de révocation.	Oui	Non	Non	Non	Non
SSLKeyRepository	Emplacement du référentiel de clés qui contient le certificat numérique de l'utilisateur, au format radical.	Oui	Non	Non	Non	Non
SSLKeyRepositoryMot de passe	Phrase passe permettant d'accéder au référentiel de clés.	Oui	Non	Non	Non	Non
SSLKeyReset Nombre	Nombre d'octets non chiffrés envoyés et reçus sur un canal SSL ou TLS avant la renégociation de la clé secrète.	Oui	Non	Non	Non	Non
Strophe TCP						

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
<u>ClntRcvBufferSize</u>	Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité client d'un canal de connexion serveur de connexion client.	Oui	Oui	Oui	Oui	Oui
<u>ClntSndBufferSize</u>	Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité client d'un canal de connexion serveur de connexion client.	Oui	Oui	Oui	Oui	Oui
<u>Connect_Timeout</u>	Nombre de secondes avant qu'une tentative de connexion du socket n'expire.	Oui	Oui	Oui	Non	Non
<u>IPAddressVersion</u>	Indique le protocole IP à utiliser pour une connexion de canal.	Oui	Non	Non	Oui	Oui
<u>KeepAlive</u>	Active ou désactive la fonction KeepAlive .	Oui	Oui	Oui	Oui	Oui

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
Windows Library1	Sous Windows uniquement, nom de la DLL de sockets TCP/IP.	Oui	Non	Non	Non	Non
Strophe Trace						
Remarque : La section Trace s'applique uniquement aux clients IBM MQ .NET et XMS .NET .						
V 9.4.0 MQDotnetTraceNiveau	Permet d'activer la trace IBM MQ .NET .	Non	Non	Non	Oui	Non
V 9.4.0 MQDotnetTraceChemin	Pointe vers un dossier dans lequel les fichiers de trace IBM MQ .NET seront créés.	Non	Non	Non	Oui	Non
V 9.4.0 MQDotnetErrorChemin	Pointe vers un dossier dans lequel les fichiers journaux d'erreurs seront créés pour la trace IBM MQ .NET .	Non	Non	Non	Oui	Non
V 9.4.0 XMSDotnetTraceNiveau	Utilisé pour activer la trace XMS .NET .	Non	Non	Non	Non	Oui
V 9.4.0 XMSDotnetTraceFilePath	Pointe vers un dossier dans lequel les fichiers de trace XMS .NET seront créés.	Non	Non	Non	Non	Oui

Tableau 16. Les attributs qui s'appliquent à chaque type de client (suite)

Nom et attributs de la section mqclient.ini	Description	C et non géré .NET	Java	JMS	.NET géré	XMS .NET géré
> V9.4.0 XMSDotnetTrace	Indique le nom de la classe à tracer pour XMS .NET.	Non	Non	Non	Non	Oui
> V9.4.0 XMSDotnetTrace	Indique la taille maximale du fichier de trace qui doit être généré pour XMS .NET.	Non	Non	Non	Non	Oui
> V9.4.0 XMSDotnetTraceFileSize	Nombre de fichiers de trace à conserver pour XMS .NET.	Non	Non	Non	Non	Oui

Section Application du fichier de configuration du client

Utilisez la section Application pour spécifier les attributs qui influencent le comportement d'équilibrage uniforme des clusters pour une application spécifique se connectant à l'aide de cette configuration. Les valeurs de cette section sont prioritaires par rapport à la section ApplicationDefaults, mais elles peuvent être remplacées par une structure MQBNO fournie via un programme.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client.](#)

Les attributs suivants peuvent être inclus dans la section Application:

Nom = ApplicationName

Identifie le nom de l'application à laquelle les options s'appliquent.

Type = Simple,ReqRep

Indique à IBM MQ le modèle général de l'activité IBM MQ à laquelle cette application participe.

BalanceTimeout = Never,Immediate,0-999999999, Valeur par défaut

Indique à IBM MQ le délai d'attente avant que l'activité de l'application puisse être interrompue pour permettre le rééquilibrage ; jamais, ou une valeur maximale de 999,999,999 secondes, avec une valeur par défaut de 10 secondes.

BalanceOptions = Aucun,IgnTrans

Aucune option d'équilibrage n'est définie ou permet l'interruption immédiate des applications actuellement impliquées dans une transaction.

Section ApplicationDefaults du fichier de configuration du client

Utilisez la section ApplicationDefaults pour spécifier les attributs qui influencent le comportement d'équilibrage uniforme des clusters par défaut pour les applications client qui se connectent à l'aide de cette configuration. Ces valeurs par défaut peuvent être remplacées par une section Application spécifique à l'application ou par une structure MQBNO fournie via un programme.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la section ApplicationDefaults :

Type = *Simple,ReqRep*

Indique à IBM MQ le modèle général de l'activité IBM MQ à laquelle cette application participe.

BalanceTimeout = *Never,Immediate,0-999999999*, Valeur par défaut

Indique à IBM MQ le délai d'attente avant que l'activité de l'application puisse être interrompue pour permettre le rééquilibrage ; jamais, ou une valeur maximale de 999,999,999 secondes, avec une valeur par défaut de 10 secondes.

BalanceOptions = *Aucun,IgnTrans*

Aucune option d'équilibrage n'est définie ou permet l'interruption immédiate des applications actuellement impliquées dans une transaction.

Strophe CHANNELS du fichier de configuration client

Utilisez la section Canaux pour spécifier des informations sur les canaux client.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la section Canaux:

CCSID = *nombre*

Numéro de jeu de caractères codés à utiliser.

Cet attribut peut être lu par les clients C, .NETnon gérés, .NETgérés et XMS .NET gérés.

Le numéro de CCSID est équivalent à la variable d'environnement [MQCCSID](#) .

ChannelDefinitionDirectory = *chemin*

Chemin de répertoire du fichier contenant la table de définition de canal du client.

Cet attribut peut être lu par les clients C, .NETnon gérés, .NETgérés et XMS .NET gérés.

Windows

Sur les systèmes Windows , la valeur par défaut est le répertoire des fichiers journaux et de données IBM MQ , généralement C:\ProgramData\IBM\MQ.

Linux

AIX

Sur les systèmes AIX and Linux , la valeur par défaut est /var/mqm.

L'annuaire ChannelDefinitionpeut contenir une URL qui fonctionne en combinaison avec l'attribut de fichier ChannelDefinition(voir «URL d'accès à la table de définition de canal du client», à la page 56).

Le chemin d'accès au répertoire ChannelDefinitionest équivalent à la variable d'environnement [MQCHLLIB](#) .

ChannelDefinitionFile = *nom_fichier|AMQCLCHL . TAB*

Nom du fichier contenant la table de définition de canal du client.

Cet attribut peut être lu par les clients C, .NETnon gérés, .NETgérés et XMS .NET gérés.

La table de définition de canal du client est équivalente à la variable d'environnement [MQCHLTAB](#) .

ReconDelay = (delay [, rand]) (delay [, rand]) ...

L'attribut ReconDelay fournit une option d'administration permettant de configurer le délai de reconnexion pour les programmes client qui peuvent se reconnecter automatiquement.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

Voici un exemple de configuration:

```
ReconDelay=(1000,200) (2000,200) (4000,1000)
```

L'exemple illustré définit un délai initial d'une seconde, plus un intervalle aléatoire pouvant atteindre 200 millisecondes. Le délai suivant est de deux secondes plus un intervalle aléatoire pouvant atteindre 200 millisecondes. Tous les retards ultérieurs sont de quatre secondes, plus un intervalle aléatoire pouvant aller jusqu'à 1000 millisecondes.

DefRecon = NO|YES|QMGR |DISABLED

L'attribut DefRecon fournit une option d'administration permettant aux programmes client de se reconnecter automatiquement ou de désactiver la reconnexion automatique d'un programme client qui a été écrit pour se reconnecter automatiquement. Vous pouvez choisir de définir ce dernier si un programme utilise une option, telle que MQPMO_LOGICAL_ORDER, qui est incompatible avec la reconnexion.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

La reconnexion automatique du client n'est pas prise en charge par IBM MQ classes for Java.

La reconnexion automatique du client dépend généralement de deux valeurs qui sont les suivantes:

- Option de reconnexion définie dans l'application MQCONN (ou la fabrique de connexions JMS)
- Option de reconnexion par défaut fournie dans toute définition de connexion client en cours d'utilisation (structure MQCD, par exemple fournie à l'aide d'un fichier CCDT).

L'attribut de fichier mqclient.ini s'applique **uniquement** si aucune définition de canal n'est utilisée pour définir l'attribut **DefReconnect** et que cette situation se comporte comme si elle avait été fournie. L'attribut **DefReconnect** du canal (et par conséquent, cet attribut, le cas échéant):

- Remplacer le code d'application si l'un ou l'autre est défini sur DISABLED
- sont remplacés par le code d'application dans tous les autres cas, si les options sont spécifiées sur MQCONN

Voir la description de [DEFRECON](#) pour une table présentant toutes les combinaisons possibles de valeurs fournies par les définitions d'application et de canal.

Remarques :

- Si un MQCD est utilisé mais qu'il est antérieur à MQCD_VERSION_10, le paramètre **DefReconnect** ne fait pas partie de la structure. Dans ce cas, la valeur de ce paramètre manquant est renseignée avec la valeur mqclient.ini file **DefReconnect** si elle est spécifiée. Cela peut se produire, par exemple, si une table de définition de canal du client au format binaire générée dans une ancienne version de IBM MQ est toujours utilisée par une application client.
- Lorsqu'elle est interprétée par le code client IBM MQ, une table de définition de canal du client JSON, voir «[Configuration d'une table de définition de canal du client au format JSON](#)», à la page 48, génère toujours des structures MQCD à la version la plus récente et fournit donc toujours la valeur par défaut (NO) pour cet attribut, sauf si elle est explicitement présente avec une valeur différente.

MQReconnectTimeout

Durée maximale, en secondes, pendant laquelle la fonction de reconnexion automatique d'un client tente de rétablir la connexion. La valeur par défaut est de 1800 secondes (30 minutes).

Cet attribut peut être lu par les clients .NET C et non gérés, ainsi que par les clients .NET gérés.

Les clients IBM MQ classes for JMS peuvent spécifier un délai de reconnexion à l'aide de la propriété de fabrique de connexions `CLIENTRECONNECTTIMEOUT`. La valeur par défaut de cette propriété est de 1800 secondes (30 minutes).

Les clients IBM MQ classes for XMS .NET peuvent spécifier un délai d'attente pour se reconnecter à l'aide des propriétés suivantes:

- Propriété de fabrique de connexions `CLIENTRECONNECTTIMEOUT`. La valeur par défaut de cette propriété est de 1800 secondes (30 minutes). Cette propriété est valide uniquement pour le mode géré.
- Propriété `XMSC.WMQ_CLIENT_RECONNECT_TIMEOUT`. La valeur par défaut de cette propriété est de 1800 secondes (30 minutes). Cette propriété est valide uniquement pour le mode géré.

Paramètres `ServerConnection`

`ServerConnection` Les paramètres sont équivalents à la variable d'environnement `MQSERVER` et indiquent l'emplacement du serveur IBM MQ et la méthode de communication à utiliser.

Cet attribut peut être lu par les clients C, .NET non gérés, .NET gérés et XMS .NET gérés.

L'attribut `ServerConnectionParms` définit uniquement un canal simple ; vous ne pouvez pas l'utiliser pour définir un canal TLS ou un canal avec des exits de canal. Il s'agit d'une chaîne au format `ChannelName/TransportType/ConnectionName`, `ConnectionName` doit être un nom de réseau qualifié complet. `ChannelName` ne peut pas contenir de barre oblique (/) car ce caractère est utilisé pour séparer le nom de canal, le type de transport et le nom de connexion.

Lorsque des paramètres `ServerConnections` sont utilisés pour définir un canal client, une longueur maximale de message de 100 Mo est utilisée. Par conséquent, la taille de message maximale en vigueur pour le canal est la valeur spécifiée dans le canal `SVRCONN` sur le serveur.

Notez qu'une seule connexion de canal client peut être établie. Par exemple, si vous avez deux entrées:

```
ServerConnectionParms=R1.SVRCONN/TCP/localhost(1963)
ServerConnectionParms=R2.SVRCONN/TCP/localhost(1863)
```

Seule la seconde est utilisée.

Spécifiez `ConnectionName` sous forme de liste de noms séparés par des virgules pour le type de transport indiqué. En règle générale, un seul nom est requis. Vous pouvez fournir plusieurs *noms d'hôte* pour configurer plusieurs connexions avec les mêmes propriétés. Les connexions sont tentées dans l'ordre dans lequel elles sont spécifiées dans la liste de connexions jusqu'à ce qu'une connexion soit établie. Si aucune connexion n'est établie, le client recommence le traitement. Les listes de connexions sont une alternative aux groupes de gestionnaires de files d'attente pour configurer les connexions pour les clients reconnectables.

Put1DefaultAlwaysSync = NO (par défaut) | YES

Contrôle le comportement de l'appel de fonction `MQPUT1` avec l'option `MQPMO_RESPONSE_AS_Q_DEF`.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

NO

Si `MQPUT1` est défini avec `MQPMO_SYNCPOINT`, il se comporte comme `MQPMO_ASYNC_RESPONSE`. De même, si `MQPUT1` est défini avec `MQPMO_NO_SYNCPOINT`, il se comporte comme `MQPMO_SYNC_RESPONSE`. Il s'agit de la valeur par défaut.

YES

`MQPUT1` se comporte comme si `MQPMO_SYNC_RESPONSE` était défini, que `MQPMO_SYNCPOINT` ou `MQPMO_NO_SYNCPOINT` soit défini.

PasswordProtection = Compatible (par défaut) |always|facultatif

Depuis la IBM MQ 8.0, les données d'authentification que les applications IBM MQ client indiquent lorsqu'elles se connectent à un gestionnaire de files d'attente peuvent être protégées à l'aide de la fonction de protection par mot de passe IBM MQ MQCSP, si la connexion n'utilise pas le chiffrement TLS.

Cet attribut peut être lu par les clients C, .NETnon gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NETgérés et XMS .NET gérés.

La protection par mot de passe MQCSP est utile à des fins de test et de développement car l'utilisation de la protection par mot de passe MQCSP est plus simple que la configuration du chiffrement TLS, mais pas sécurisée.

Pour plus d'informations sur la protection des données d'identification dans la structure MQCSP et sur les valeurs pouvant être définies pour cet attribut, voir [Protection par mot de passe MQCSP](#).

Tâches associées

[Connexion des applications IBM MQ MQI aux gestionnaires de files d'attente](#)

Multi Section de chemin ClientExitdu fichier de configuration du client

Utilisez la section de chemin ClientExitpour spécifier les emplacements par défaut des exits de canal sur le client.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la section ClientExitPath:

ExitsDefaultPath = chaîne

Indique l'emplacement des exits de canal 32 bits pour les clients.

Cet attribut peut être lu par les clients C, .NETnon gérés, .NETgérés, XMS .NETgérés, IBM MQ classes for Javaet IBM MQ classes for JMS . Les clients IBM MQ classes for Java et IBM MQ classes for JMS utilisent cet attribut pour localiser les exits de canal 32 bits qui ne sont pas écrits dans Java.

ExitsDefaultPath64 = chaîne

Indique l'emplacement des exits de canal 64 bits pour les clients.

Cet attribut peut être lu par les clients C, .NETnon gérés, .NETgérés, XMS .NETgérés, IBM MQ classes for Javaet IBM MQ classes for JMS . Les clients IBM MQ classes for Java et IBM MQ classes for JMS utilisent cet attribut pour localiser les exits de canal 64 bits qui ne sont pas écrits dans Java.

JavaExitsClassPath = chaîne

Valeurs à ajouter au chemin d'accès aux classes lors de l'exécution d'un exit Java . Elle est ignorée par les exits dans n'importe quelle autre langue.

Cet attribut peut être lu par les clients IBM MQ classes for Java et IBM MQ classes for JMS .

Dans le fichier de configuration JMS , le nom de chemin JavaExitsClassest associé à la valeur standard com.ibm.mq.cfg. et ce nom complet est également utilisé sur la propriété système IBM MQ .

Multi Section de connexion du fichier de configuration du client

Utilisez la section de connexion pour spécifier un nom d'application.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

L'attribut suivant peut être inclus dans la section Connexion:

ApplName = ExampleAppName

Vous pouvez spécifier un nom d'application dans le fichier de configuration du client.

Cet attribut peut être utilisé par les clients .NET C et non gérés.

Multi

Section JMQUI du fichier de configuration du client

Utilisez la section JMQUI pour spécifier les paramètres de configuration de l'interface JMQUI (Java Message Queuing Interface) utilisée par IBM MQ classes for Java et IBM MQ classes for JMS.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

L'attribut suivant peut être inclus dans la strophe JMQUI:

useMQCSPauthentication = NON|OUI

Contrôle si les applications IBM MQ classes for Java et IBM MQ classes for JMS doivent utiliser le mode compatibilité ou le mode d'authentification MQCSP lors de l'authentification avec un gestionnaire de files d'attente.

Cet attribut peut être lu par les clients IBM MQ classes for Java et IBM MQ classes for JMS .

Cet attribut peut avoir les valeurs suivantes :

NO

Utilisez le mode compatibilité lors de l'authentification avec un gestionnaire de files d'attente. Il s'agit de la valeur par défaut dans les versions antérieures à IBM MQ 9.3.0.

YES

Utilisez le mode d'authentification MQCSP lors de l'authentification avec un gestionnaire de files d'attente. Il s'agit de la valeur par défaut de IBM MQ 9.3.0.

Il existe plusieurs autres façons de définir le mode d'authentification qui est prioritaire sur la valeur de l'attribut **useMQCSPauthentication** . Pour plus d'informations sur le mode compatibilité et le mode d'authentification MQCSP, voir [Authentification de connexion avec le client Java](#).

Windows

Sections LU62, NETBIOS et SPX du fichier de configuration client

Sur les systèmes Windows uniquement, utilisez ces sections pour spécifier des paramètres de configuration pour les protocoles réseau spécifiés.

Section LU62

Utilisez la section LU62 pour spécifier les paramètres de configuration du protocole SNA LU 6.2 . Les attributs suivants peuvent être inclus dans cette section:

Library1 = DLLName|WCPIC32

Nom de la DLL APPC.

Library2 = DLLName|WCPIC32

Identique à Library1, utilisé si le code est stocké dans deux bibliothèques distinctes.

TPName

Nom du programme transactionnel à démarrer sur le site distant.

Section NETBIOS

Utilisez la section NETBIOS pour spécifier les paramètres de configuration du protocole NetBIOS . Les attributs suivants peuvent être inclus dans cette section:

AdapterNum = nombre|0

Numéro de la carte de réseau local.

Library1 = DLLName|NETAPI32

Nom de la DLL NetBIOS .

LocalName = nom

Nom sous lequel cet ordinateur est connu sur le réseau local.

Cette valeur est équivalente à la variable d'environnement MQNAME .

NumCmds = nombre|1

Nombre de commandes à allouer.

NumSess = nombre|1

Nombre de sessions à allouer.

Section SPX

Utilisez la section SPX pour spécifier les paramètres de configuration du protocole SPX. Les attributs suivants peuvent être inclus dans cette section:

BoardNum = nombre|0

Numéro de carte de réseau local.

KeepAlive = YES|NO

Activez ou désactivez la fonction KeepAlive .

KeepAlive = YES permet à SPX de vérifier périodiquement que l'autre extrémité de la connexion est toujours disponible. Si tel n'est pas le cas, le canal est fermé.

Library1 = DLLName|WSOCK32 . DLL

Nom de la DLL SPX.

Library2 = DLLName|WSOCK32 . DLL

Identique à Library1, utilisé si le code est stocké dans deux bibliothèques distinctes.

Socket = nombre|5E86

Nombre de sockets SPX en notation hexadécimale.

Multi

Section MessageBuffer du fichier de configuration du client

Utilisez la section MessageBuffer pour spécifier des informations sur les mémoires tampon de messages.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la section MessageBuffer :

MaximumSize = entier|1

Taille, en kilooctets, de la mémoire tampon de lecture anticipée, comprise entre 1 et 999 999.

Cet attribut peut être lu par les clients C, .NETnon gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NETgérés et XMS .NET gérés.

Les valeurs spéciales suivantes existent:

-1

Le client détermine la valeur appropriée.

0

La lecture anticipée est désactivée pour le client.

PurgeTime = entier|600

Intervalle, en secondes, après lequel les messages laissés dans la mémoire tampon de lecture anticipée sont purgés.

Cet attribut peut être lu par les clients C, .NETnon gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NETgérés et XMS .NET gérés.

Si l'application client sélectionne des messages en fonction de MsgId ou CorrelId , il est possible que la mémoire tampon de lecture anticipée contienne des messages envoyés au client avec un MsgId ou CorrelIdprécédemment demandé. Ces messages sont ensuite bloqués dans la mémoire tampon de lecture anticipée jusqu'à ce qu'une instruction MQGET soit émise avec un MsgId ou CorrelIdapproprié. Vous pouvez purger les messages de la mémoire tampon de lecture anticipée en définissant PurgeTime. Tous les messages qui sont restés dans la mémoire tampon de lecture

anticipée plus longtemps que l'intervalle de purge sont automatiquement purgés. Ces messages ont déjà été supprimés de la file d'attente du gestionnaire de files d'attente. Par conséquent, à moins qu'ils ne soient consultés, ils sont perdus.

La plage valide est comprise entre 1 et 999 999 secondes, ou la valeur spéciale 0, ce qui signifie qu'aucune purge n'a lieu.

UpdatePercentage = entier|-1

Valeur de pourcentage de mise à jour, comprise entre 1 et 100, utilisée pour calculer la valeur de seuil afin de déterminer à quel moment une application client effectue une nouvelle demande auprès du serveur. La valeur spéciale -1 indique que le client détermine la valeur appropriée.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

Le client envoie périodiquement une demande au serveur indiquant la quantité de données consommée par l'application client. Une demande est envoyée lorsque le nombre d'octets, n , extraits par le client via des appels MQGET dépasse un seuil T . n est réinitialisé à zéro chaque fois qu'une nouvelle demande est envoyée au serveur.

Le seuil T est calculé comme suit:

$$T = \text{Upper} - \text{Lower}$$

La valeur supérieure correspond à la taille de la mémoire tampon de lecture anticipée, spécifiée par l'attribut *MaximumSize*, en kilooctets. Sa valeur par défaut est 100 Ko.

La valeur inférieure est inférieure à la valeur supérieure et est spécifiée par l'attribut *UpdatePercentage*. Cet attribut est un nombre compris entre 1 et 100 et a une valeur par défaut de 20. La valeur la plus basse est calculée comme suit:

$$\text{Lower} = \text{Upper} \times \text{UpdatePercentage} / 100$$

Exemple 1 :

Les attributs *MaximumSize* et *UpdatePercentage* prennent leurs valeurs par défaut de 100 Ko et 20 Ko.

Le client appelle MQGET pour extraire un message, et le fait à plusieurs reprises. Cette opération se poursuit jusqu'à ce que MQGET ait consommé n octets.

Utilisation du calcul

$$T = \text{Upper} - \text{Lower}$$

T est $(100-20) = 80$ Ko.

Ainsi, lorsque des appels MQGET ont supprimé 80 Ko d'une file d'attente, le client effectue automatiquement une nouvelle demande.

Exemple 2 :

Les attributs *MaximumSize* prennent la valeur par défaut de 100 Ko et la valeur 40 est choisie pour *UpdatePercentage*.

Le client appelle MQGET pour extraire un message, et le fait à plusieurs reprises. Cette opération se poursuit jusqu'à ce que MQGET ait consommé n octets.

Utilisation du calcul

$$T = \text{Upper} - \text{Lower}$$

T est $(100-40) = 60$ Ko

Ainsi, lorsque des appels MQGET ont supprimé 60 Ko d'une file d'attente, le client effectue automatiquement une nouvelle demande. C'est plus tôt que dans l'EXEMPLE 1 où les valeurs par défaut ont été utilisées.

Par conséquent, le choix d'un seuil plus élevé T a tendance à réduire la fréquence à laquelle les demandes sont envoyées du client au serveur. Inversement, le choix d'un seuil plus petit T a tendance à augmenter la fréquence des demandes envoyées du client au serveur.

Toutefois, le choix d'un seuil élevé T peut signifier que le gain de performances de la lecture anticipée est réduit, car le risque de vidage de la mémoire tampon de lecture anticipée peut augmenter. Lorsque cela se produit, un appel MQGET peut devoir être mis en pause, en attendant que les données arrivent du serveur.

Multi

Section PreConnect du fichier de configuration du client

Utilisez la section PreConnect pour configurer l'exit PreConnect dans le fichier `mqclient.ini`.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la section PreConnect :

Data = données_utilisateur

Cet attribut spécifie les données utilisateur qui sont transmises à l'exit de préconnexion. Les données transmises à l'exit de préconnexion sont spécifiques à l'implémentation de l'exit de préconnexion que vous utilisez et aux données qu'il s'attend à transmettre.

Cet attribut peut être lu par les clients .NET C et non gérés.

Par exemple, cet attribut peut être utilisé pour spécifier l'URL du référentiel dans lequel les définitions de connexion sont stockées, par exemple, lors de l'utilisation d'un serveur LDAP:

```
Data = ldap://myLDAPServer.com:389/cn=wmq,ou=ibm,ou=com
```

Fonction = myFunc

Nom du point d'entrée fonctionnel dans la bibliothèque qui contient le code d'exit PreConnect .

Cet attribut peut être lu par les clients .NET C et non gérés.

La définition de fonction est conforme au prototype d'exit PreConnect [MQ_PRECONNECT_EXIT](#).

La longueur maximale de cette zone est MQ_EXIT_NAME_LENGTH.

Module = myMod

Nom du module contenant le code d'exit d'API.

Cet attribut peut être lu par les clients .NET C et non gérés.

Si cette zone contient le nom de chemin d'accès complet du module, elle est utilisée telle qu'elle est.

Sequence = numéro_séquence

Séquence dans laquelle cet exit est appelé par rapport à d'autres exits. Un exit avec un numéro de séquence faible est appelé avant un exit avec un numéro de séquence plus élevé. Il n'est pas nécessaire que la numérotation séquentielle des sorties soit continue ; une séquence de 1, 2, 3 a le même résultat qu'une séquence de 7, 42, 1096. Cet attribut est une valeur numérique non signée.

Cet attribut peut être lu par les clients .NET C et non gérés.

Plusieurs sections PreConnect peuvent être définies dans le fichier `mqclient.ini`. L'ordre de traitement de chaque exit est déterminé par l'attribut Séquence de la section.

Tâches associées

[Référencement des définitions de connexion à l'aide d'un exit de préconnexion à partir d'un référentiel](#)

Section de sécurité du fichier de configuration du client

Utilisez la section Sécurité pour désactiver ou activer AMS pour les connexions client à un gestionnaire de files d'attente.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

L'attribut suivant peut être inclus dans la section Sécurité:

DisableClientAMS = NO|OUI

L'attribut DisableClientAMS vous permet de désactiver IBM MQ Advanced Message Security (AMS) si vous utilisez un client IBM MQ pour vous connecter à un gestionnaire de files d'attente à partir d'une version antérieure du produit et qu'une erreur 2085 (MQRC_UNKNOWN_OBJECT_NAME) est signalée.

IBM MQ Advanced Message Security (AMS) étant automatiquement activé dans un client IBM MQ, le client tente par défaut de vérifier les règles de sécurité des objets dans le gestionnaire de files d'attente.

Les exemples suivants montrent comment utiliser l'attribut DisableClientAMS :

- Pour désactiver AMS:

```
Security:
DisableClientAMS=Yes
```

- Pour activer la AMS :

```
Security:
DisableClientAMS=No
```

Cet attribut peut être lu par les clients C, IBM MQ classes for Java et IBM MQ classes for JMS .

MQIInitialKeyFile = *nom_chemin*

Chemin d'accès complet et nom du fichier contenant la clé initiale qui a été utilisée pour chiffrer les données d'identification fournies par le client. La clé initiale doit être spécifiée si un fichier de clés initial a été spécifié lorsque la phrase passe du référentiel de clés a été chiffrée à l'aide de l'utilitaire **runmqicred** .

Cet attribut peut être lu par C et par les clients .NET non gérés.

Tâches associées

[Désactivation d'Advanced Message Security sur le client](#)

Strophe SSL du fichier de configuration client

Utilisez la section SSL pour spécifier des informations sur l'utilisation de TLS.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la strophe SSL:

OutboundSNI = CHANNEL | NOM_HÔTE

Si **OutboundSNI** est défini sur CHANNEL, les clients compatibles SNI définissent SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS.

Si cet attribut est défini sur HOSTNAME, les clients compatibles SNI définissent l'en-tête SNI sur le nom d'hôte, ce qui entraîne la réception par les demandes de connexion sortantes du certificat par défaut du gestionnaire de files d'attente éloignées lors de l'établissement de liaison TLS. Par conséquent, les certificats par canal ne peuvent pas être utilisés.

Cet attribut peut être lu par les clients C, .NET, IBM MQ classes for Java et IBM MQ classes for JMS non gérés.

Les valeurs de propriété étant interprétées avec sensibilité à la casse par le client Java/JMS , les valeurs YES/NO doivent être définies en majuscules.

Depuis la IBM MQ 9.3.0, le client IBM MQ géré .NET a été mis à jour pour définir SERVERNAME sur le nom d'hôte respectif si la propriété **OutboundSNI** est définie sur HOSTNAME, ce qui permet à un client IBM MQ géré .NET de se connecter à un gestionnaire de files d'attente à l'aide de [Red Hat OpenShift routes](#).

Remarque : Si une application avec le paramètre **OutboundSNI** HOSTNAME se connecte à un canal avec un libellé de certificat configuré, l'application est rejetée avec une erreur MQRC_SSL_INITIALIZATION_ERROR et un message AMQ9673 est imprimé dans les journaux des erreurs du gestionnaire de files d'attente.

AllowOutboundSNI = YES (valeur par défaut) | NON

Si cette option est activée, les clients compatibles SNI définiront SNI sur le nom de canal IBM MQ cible sur le système distant lors du lancement d'une connexion TLS. Si cet attribut est défini sur NO, les clients compatibles SNI ne définissent pas l'en-tête SNI, ce qui entraîne la réception par les demandes de connexion sortantes du certificat par défaut du gestionnaire de files d'attente éloignées lors de l'établissement de liaison TLS. Par conséquent, les certificats par canal ne peuvent pas être utilisés.

Cet attribut peut être lu par les clients C, .NET, IBM MQ classes for Java et IBM MQ classes for JMS non gérés.

Les valeurs de propriété étant interprétées avec sensibilité à la casse par le client Java/JMS , les valeurs YES/NO doivent être définies en majuscules.



Avertissement : Deprecated Depuis la IBM MQ 9.3.0 , la propriété **AllowOutboundSNI** est obsolète et n'est disponible qu'à des fins de compatibilité avec les versions antérieures.

AllowOutboundSNI défini sur YES fournit la même fonction que **OutboundSNI** défini sur CHANNEL, tandis que **AllowOutboundSNI** défini sur NO fournit la même fonction que **OutboundSNI** défini sur HOSTNAME.

Si les attributs **AllowOutboundSNI** et **OutboundSNI** sont présents dans la strophe SSL, le paramètre **OutboundSNI** est prioritaire.

IBM I **ALW** **AllowTLSV13 =Y | YES | T | TRUE (valeur par défaut) | N | NO | F | FALSE**

Indique si un gestionnaire de files d'attente peut utiliser les CipherSpecs TLS 1.3 CipherSpecs (voir [Activation des CipherSpecs CipherSpecs](#)).

Cet attribut peut être lu par les clients C/C + +.

Cet attribut comporte les valeurs possibles suivantes :

- Y (valeur par défaut), YES (valeur par défaut), T (valeur par défaut) ou TRUE (valeur par défaut): Active TLS 1.3 qui permet au gestionnaire de files d'attente d'utiliser les CipherSpecs TLS 1.3 CipherSpecs.
- N, NO, Fou FALSE: désactive TLS 1.3, ce qui signifie que le gestionnaire de files d'attente ne peut pas utiliser TLS 1.3 CipherSpecs.

Remarque : Lors de l'utilisation du client MQI, la valeur de **AllowTLSV13** est déduite sauf si elle est explicitement spécifiée dans la strophe SSL du fichier «Strophe SSL du fichier de configuration client», à la page 201 utilisé par l'application. Pour plus d'informations, voir [IBM MQ MQI Client and TLS 1.3](#).

CDPCheckExtensions = YES|NO (valeur par défaut)

CDPCheckExtensions indique si les canaux TLS de ce gestionnaire de files d'attente tentent de vérifier les serveurs CDP nommés dans les extensions de certificat de point CrLDistribution.

Cet attribut peut être lu par les clients .NET C et non gérés.

Cet attribut comporte les valeurs possibles suivantes :

- YES (valeur par défaut): les canaux TLS tentent de vérifier les serveurs CDP pour déterminer si un certificat numérique est révoqué.
- NON: Les canaux TLS n'essaient pas de vérifier les serveurs CDP. Cette valeur est la valeur par défaut.

CertificateLabel = chaîne

Libellé de certificat de la définition de canal.

Cet attribut peut être lu par les clients .NET C et non gérés.

Pour plus d'informations, voir [Label de certificat \(CERTLABEL\)](#) .

CertificateValPolicy = chaîne

Détermine le type de validation de certificat utilisé.

Cet attribut peut être lu par les clients .NET C et non gérés.

Cet attribut comporte les valeurs possibles suivantes :

ANY

Utilisez les règles de validation de certificat prises en charge par la bibliothèque de sockets sécurisés sous-jacente. Il s'agit du paramètre par défaut.

RFC5280

Utilisez uniquement la validation de certificat qui est conforme à la norme RFC 5280.



N'utilisez aucune validation de certificat.

ClientRevocationChecks = REQUIRED|FACULTATIF|DISABLED

Détermine comment la vérification de la révocation de certificat est configurée si l'appel de connexion client utilise un canal TLS. Voir aussi [OCSPAuthentication](#).

Cet attribut peut être lu par les clients .NET C et non gérés.

Cet attribut comporte les valeurs possibles suivantes :

REQUIRED (valeur par défaut)

Tente de charger la configuration de révocation de certificat à partir de la table de définition de canal du client et d'effectuer la vérification de révocation telle que configurée. Si le fichier CCDT ne peut pas être ouvert ou qu'il n'est pas possible de valider le certificat (car un serveur OCSP ou CRL n'est pas disponible, par exemple), l'appel MQCONN échoue. Aucune vérification de révocation n'est effectuée si la table de définition de canal du client ne contient pas de configuration de révocation, mais cela n'entraîne pas l'échec du canal.

Windows Sur les systèmes Windows , vous pouvez également utiliser Active Directory pour la vérification de la révocation de la liste de révocation de certificat. Vous ne pouvez pas utiliser Active Directory pour la vérification de la révocation OCSP.

Si vous utilisez MQSCO ou CCDT, la connexion aboutit. S'il n'existe pas de fichier CCDT et si MQSCO n'est pas non plus fourni, la connexion échoue avec le code anomalie 2059 et le journal des erreurs signale AMQ9518E: Fichier '/var/mqm/AMQCLCHL.TAB' introuvable.

Facultatif

Comme pour REQUIRED, mais s'il n'est pas possible de charger la configuration de révocation de certificat, le canal n'échoue pas.

DEACTIVE

Aucune tentative n'est effectuée pour charger la configuration de révocation de certificat à partir de la table de définition de canal du client et aucune vérification de révocation de certificat n'est effectuée.

Remarque : Si vous utilisez MQCONNX plutôt que des appels MQCONN, vous pouvez choisir de fournir des enregistrements d'informations d'authentification (MQAIR) via MQSCO. Le comportement par défaut avec MQCONNX est donc de ne pas échouer si le fichier CCDT ne peut pas être ouvert, mais de supposer que vous fournissez un MQAIR (même si vous choisissez de ne pas le faire).

EncryptionPolicySuiteB = chaîne

Détermine si un canal utilise la cryptographie compatible Suite-B et le niveau de puissance à utiliser.

Cet attribut peut être lu par les clients .NET C et non gérés.

Cet attribut comporte les valeurs possibles suivantes :

Aucun

La cryptographie compatible Suite-B n'est pas utilisée. Il s'agit du paramètre par défaut.

128_BIT,192_BIT

Définit la force de sécurité sur les niveaux 128 bits et 192 bits.

128_BIT

Définit le niveau de sécurité à 128 bits.

192_BIT

Définit le niveau de sécurité sur 192 bits.

ALW

EnvironmentScope=PROCESSUS|CONNEXION

Contrôle si IBM MQ utilise un environnement IBM Global Security Kit (GSKit) unique pour l'ensemble du processus ou un environnement GSKit par connexion.

Cet attribut peut être lu par les clients C.

Cet attribut comporte les valeurs possibles suivantes :

PROCESS

Un environnement GSKit unique est utilisé pour plusieurs connexions créées par le processus. L'utilisation de ce paramètre signifie que les modifications du magasin de clés TLS ne seront pas disponibles tant que toutes les connexions TLS actives au sein du processus ne seront pas arrêtées.

Cette valeur est la valeur par défaut.

CONNECTION

Un environnement GSKit est créé pour chaque connexion au sein du même processus. L'activation de cette option signifie que les modifications apportées au magasin de clés TLS seront prises en compte immédiatement par toutes les nouvelles connexions TLS démarrées par le processus.



Avertissement : L'activation de ce mode de fonctionnement permet aux applications d'utiliser des ressources d'UC et de mémoire supplémentaires pour créer chaque environnement GSKit . Cette consommation de ressources augmente avec chaque connexion TLS simultanée supplémentaire.

ALW

MinimumRSAKeySize=int

Indique la taille de clé minimale que les certificats RSA doivent avoir pour être acceptés. Autorise toute valeur égale ou supérieure à 0. La valeur par défaut est 1 si elle n'est pas spécifiée.

Cet attribut peut être lu par les clients C/C + +.

OCSPAAuthentication = FACULTATIF|OBLIGATOIRE|AVERTISSEMENT

Définit le comportement de IBM MQ lorsque OCSP est activé et que la vérification de révocation OCSP ne parvient pas à déterminer le statut de révocation de certificat. Voir aussi **ClientRevocationChecks**.

Cet attribut peut être lu par les clients .NET C et non gérés.

Cet attribut comporte les valeurs possibles suivantes :

Facultatif

Tout certificat dont le statut de révocation ne peut pas être déterminé par la vérification OCSP est accepté et aucun message d'avertissement ou d'erreur n'est généré. La connexion SSL ou TLS se poursuit comme si aucune vérification de révocation n'avait été effectuée.

REQUIRED

La vérification OCSP doit générer un résultat de révocation définitif pour chaque certificat SSL ou TLS vérifié. Tout certificat SSL ou TLS dont le statut de révocation ne peut pas être vérifié est rejeté avec un message d'erreur. Si les messages d'événement SSL du gestionnaire de files d'attente sont activés, un message MQRC_CHANNEL_SSL_ERROR avec un ReasonQualifier de MQRQ_SSL_HANDSHAKE_ERROR est généré. La connexion est fermée.

Cette valeur est la valeur par défaut.

AVERTISSEMENT

Un avertissement est signalé dans les journaux d'erreurs du gestionnaire de files d'attente si une vérification de révocation OCSP ne parvient pas à déterminer le statut de révocation d'un certificat SSL ou TLS. Si les messages d'événement SSL du gestionnaire de files d'attente sont activés, un message MQRC_CHANNEL_SSL_WARNING avec un ReasonQualifier de MQRQ_SSL_UNKNOWN_REVOCATION est généré. La connexion est autorisée à se poursuivre.

OCSPCheckExtensions = YES|NO

Contrôle si IBM MQ agit sur les extensions de certificat d'accès AuthorityInfo.

Cet attribut peut être lu par les clients .NET C et non gérés.

Si la valeur est définie sur NO, IBM MQ ignore les extensions de certificat d'accès AuthorityInfo et ne tente pas de vérifier la sécurité OCSP. La valeur par défaut est YES.

ALW

OCSPTimeout = *nombre*

Nombre de secondes d'attente d'un répondeur OCSP lors de l'exécution d'une vérification de révocation.

Cet attribut peut être lu par les clients .NET C et non gérés.

Depuis la IBM MQ 9.3.0, si la valeur 0 est définie, le délai d'attente par défaut de 30 secondes est utilisé.

Si aucune valeur n'est définie, la valeur par défaut IBM MQ de 30 secondes est utilisée.

ALW

PeerCertChainValidation=*chaîne*

Cet attribut peut être lu par C et par les clients .NET non gérés.

La chaîne peut être l'une des deux valeurs suivantes:

- Usepeerchain [**Valeur par défaut**]: La chaîne de certificats fournie par l'homologue peut être utilisée pour combler les lacunes de la chaîne de confiance lors de la validation des certificats. A l'exception du certificat racine.
- Truststoreonly [**Non recommandé**]: seuls les certificats du magasin de clés de confiance seront utilisés pour valider le certificat de l'homologue.

SSLCryptoHardware = *chaîne*

Définit la chaîne de paramètres requise pour configurer le matériel de cryptographie PKCS #11 présent sur le système.

Cet attribut peut être lu par les clients .NET C et non gérés.

Spécifiez une chaîne au format suivant: GSK_PKCS11 = *driver path and filename;token label;token password;symmetric cipher setting*;

Par exemple: GSK_PKCS11=/usr/lib/pkcs11/
PKCS11_API.so;tokenlabel;password;SYMMETRIC_CIPHER_ON

Le chemin du pilote est un chemin d'accès absolu à la bibliothèque partagée fournissant la prise en charge de la carte PKCS #11. Le nom du fichier de pilote est le nom de la bibliothèque partagée. Exemple de valeur requise pour le chemin d'accès au pilote PKCS #11 et le nom de fichier: /usr/lib/pkcs11/PKCS11_API.so. Pour accéder aux opérations de chiffrement symétrique via GSKit, spécifiez le paramètre de définition de chiffrement symétrique. La valeur de ce paramètre est l'une des suivantes:

SYMMETRIC_CIPHER_OFF

N'accédez pas aux opérations de chiffrement symétrique. Il s'agit du paramètre par défaut.

SYMMETRIC_CIPHER_ON

Accès aux opérations de chiffrement symétrique.

Linux **AIX** Lorsque vous fournissez les différents composants de la chaîne, vous devez mettre en échappement les caractères point-virgule à l'aide de la barre oblique inversée, car le point-virgule est traité comme un commentaire. Par exemple: '\;'

Vous devez protéger le mot de passe de jeton contenu dans la chaîne d'attribut

SSLCryptoHardware . Pour plus d'informations, voir [IBM MQ clients using cryptographic hardware](#) .

Pour gérer les mots de passe chiffrés, la longueur de la chaîne n'est plus limitée.

Par défaut, cette zone reste vide. Si vous spécifiez une chaîne dont le format n'est pas correct, une erreur est générée.

SSLCryptoHardwareKeyFile = nom_chemin

Chemin d'accès complet et nom du fichier contenant la clé initiale qui a été utilisée pour chiffrer le mot de passe dans la chaîne de configuration du matériel de cryptographie PKCS #11 spécifiée avec l'attribut **SSLCryptoHardware** . La clé initiale doit être spécifiée si un fichier de clés initial a été spécifié lorsque le mot de passe dans la chaîne de configuration du matériel de cryptographie a été chiffré à l'aide de la commande **runp11cred** . Pour plus d'informations, voir [Clients IBM MQ utilisant du matériel de cryptographie](#).

Cet attribut peut être lu par C et par les clients .NET non gérés.

SSLFipsRequired = OUI|NON

Indique si seuls les algorithmes certifiés FIPS doivent être utilisés si la cryptographie est effectuée dans IBM MQ.

Cet attribut peut être lu par C et par les clients .NET non gérés.

Si le matériel de cryptographie est configuré, les modules de cryptographie utilisés sont ceux fournis par le produit matériel. Ils peuvent ou non être certifiés FIPS à un niveau particulier, en fonction du produit matériel utilisé.

SSLHTTPProxyName = chaîne

La chaîne correspond au nom d'hôte ou à l'adresse réseau du serveur proxy HTTP qui doit être utilisé par GSKit pour les vérifications OCSP. Cette adresse peut être suivie d'un numéro de port facultatif, entre parenthèses. Si vous n'indiquez pas le numéro de port, le port HTTP par défaut, 80, est utilisé.

Cet attribut peut être lu par C et par les clients .NET non gérés.

AIX Pour 32 bits sur AIX, l'adresse réseau ne peut être qu'une adresse IPv4 .

Sur les autres plateformes, l'adresse réseau peut être une adresse IPv4 ou IPv6 .

Cet attribut peut être nécessaire si, par exemple, un pare-feu empêche l'accès à l'URL du répondeur OCSP.

ALW **SSLHTTPConnectTimeout = nombre|0**

Nombre de secondes d'attente pour l'établissement d'une connexion réseau à un serveur HTTP lors de l'exécution d'une vérification de révocation.

Cet attribut peut être lu par les clients .NET C et non gérés.

Si aucune valeur n'est définie, la valeur par défaut IBM MQ 0 (off) est utilisée.

SSLKeyRepository = nom_chemin

Chemin d'accès complet et nom de fichier du référentiel de clés qui contient le certificat numérique de l'utilisateur. Si l'extension de fichier n'est pas spécifiée, elle est supposée être .kdb.

Cet attribut peut être lu par C et par les clients .NET non gérés.

SSLKeyRepositoryPassword = phrase passe

Phrase passe permettant d'accéder au référentiel de clés. La valeur peut être une chaîne de texte en clair ou une phrase passe chiffrée à l'aide de l'utilitaire **runmqicred**.

Cet attribut peut être lu par C et par les clients .NET non gérés.

SSLKeyResetNombre = entier|0

Nombre d'octets non chiffrés envoyés et reçus sur un canal TLS avant la renégociation de la clé secrète.

Cet attribut peut être lu par C et par les clients .NET non gérés.

La valeur doit être comprise entre 0 et 999999999.

La valeur par défaut est 0, ce qui signifie que les clés secrètes ne sont jamais renégociées.

Si vous indiquez une valeur comprise entre 1 et 32768, les canaux TLS utilisent un nombre de réinitialisations de clé secrète de 32768 (32Kb). Cela permet d'éviter un nombre excessif de réinitialisations de clé, ce qui se produit pour les valeurs de réinitialisation de clé secrète de petite taille.

Multi

Section TCP du fichier de configuration du client

Utilisez la section TCP pour spécifier les paramètres de configuration du protocole de réseau TCP.

Remarque : La description de chaque attribut de cette strophe indique quels clients IBM MQ peuvent lire cet attribut. Pour une table récapitulative de toutes les strophes du fichier de configuration IBM MQ MQI client, voir [Quels sont les attributs IBM MQ qui peuvent être lus par chaque client](#).

Les attributs suivants peuvent être inclus dans la strophe TCP:

ClntRcvBuffSize = nombre|0

Taille en octets de la mémoire tampon de réception TCP/IP utilisée par l'extrémité client d'un canal de connexion serveur de connexion client.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

La valeur zéro indique que le système d'exploitation va gérer les tailles de mémoire tampon, par opposition aux tailles de mémoire tampon fixées par IBM MQ. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ, 32768, est utilisée.

ClntSndBuffSize = nombre|0

Taille en octets de la mémoire tampon d'envoi TCP/IP utilisée par l'extrémité client d'un canal de connexion serveur de connexion client.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

La valeur zéro indique que le système d'exploitation va gérer les tailles de mémoire tampon, par opposition aux tailles de mémoire tampon fixées par IBM MQ. Si la valeur est définie sur zéro, les valeurs par défaut du système d'exploitation sont utilisées. Si aucune valeur n'est définie, la valeur par défaut IBM MQ, 32768, est utilisée.

Connect_Timeout = nombre

Nombre de secondes avant qu'une tentative de connexion du socket n'expire.

Si **ConnectTimeout** = 0 et que SOCK_NONBLOCK est émis avant un appel de connexion asynchrone (), l'appel n'est pas bloqué. La valeur de délai d'attente par défaut de 20 secondes (CONNECT_WAIT_MAX) est applicable pour la vérification du statut de connexion.

Cet attribut peut être lu par les clients C, .NET, IBM MQ classes for Java et IBM MQ classes for JMS non gérés.

Les processus de canal IBM MQ se connectent via des sockets non bloquants. Par conséquent, si l'autre extrémité du socket n'est pas prête, connect () renvoie immédiatement *EINPROGRESS* ou *EWOULDBLOCK*. Par la suite, aucune tentative de reconnexion n'est effectuée.

Si Connect_Timeout est défini sur une valeur différente de zéro, IBM MQ attend la période stipulée sur l'appel select () pour que le socket soit prêt. Cela augmente les chances de réussite d'un appel connect () ultérieur. Cette option peut être utile dans les cas où les connexions nécessitent un délai d'attente, en raison d'une charge élevée sur le réseau.

Il n'existe aucune relation entre les paramètres Connect_Timeout, ClntSndBuffSize et ClntRcvBuffSize .

IPAddressVersion = MQIPADDR_IPV4|MQIPADDR_IPV6

Indique le protocole IP à utiliser pour une connexion de canal.

Cet attribut peut être lu par les clients C, .NET non gérés, .NET gérés et XMS .NET gérés.

Il peut avoir les valeurs de chaîne MQIPADDR_IPV4 ou MQIPADDR_IPV6. Ces valeurs ont la même signification que IPV4 et IPV6 dans **ALTER QMGR IPADDRV** et la variable d'environnement **MQIPADDRV** .

KeepAlive = YES|NO

Activez ou désactivez la fonction KeepAlive . KeepAlive=YES permet à TCP/IP de vérifier périodiquement que l'autre extrémité de la connexion est toujours disponible. Si tel n'est pas le cas, le canal est fermé.

Cet attribut peut être lu par les clients C, .NET non gérés, IBM MQ classes for Java, IBM MQ classes for JMS, .NET gérés et XMS .NET gérés.

Windows Library1 = DLLName|WSOCK32

(Windows uniquement) Nom de la DLL de sockets TCP/IP.

Cet attribut peut être lu par les clients .NET C et non gérés.

V 9.4.0 Section de trace du fichier de configuration du client

Utilisez la strophe Trace pour activer la trace pour les bibliothèques client IBM MQ .NET et XMS .NET .

Les attributs suivants peuvent être inclus dans la strophe TRACE:

MQDotnetTraceLevel=0 (valeur par défaut) |1|2

L'attribut **MQDotnetTraceLevel1** est utilisé pour démarrer ou arrêter la trace IBM MQ .NET :

- 0: arrête le traçage-il s'agit de la valeur par défaut.
- 1: démarre le traçage avec moins de détails.
- 2: Commence le traçage avec tous les détails-recommandé.

Cet attribut peut être lu par le client IBM MQ .NET géré.

MQDotnetTracePath =nom_chemin

L'attribut **MQDotnetTracePath** pointe vers un dossier dans lequel les fichiers de trace IBM MQ .NET seront créés. Le répertoire en cours de l'application est utilisé si le chemin est vide ou si la propriété **MQDotnetTracePath** n'est pas définie.

Cet attribut peut être lu par le client IBM MQ .NET géré.

MQDotnetErrorPath =nom_chemin

L'attribut **MQDotnetErrorPath** pointe vers un dossier dans lequel les fichiers journaux d'erreurs seront créés pour la trace IBM MQ .NET . Le répertoire de travail de l'application est utilisé si le chemin est vide ou si l'attribut **MQDotnetErrorPath** n'est pas défini.

Cet attribut peut être lu par le client IBM MQ .NET géré.

XMSDotnetTraceLevel=0 (valeur par défaut) |1|2

L'attribut **XMSDotnetTraceLevel1** est utilisé pour démarrer ou arrêter la trace XMS .NET :

- 0: arrête le traçage-il s'agit de la valeur par défaut.
- 1: Démarre le traçage avec le format de base.
- 2: démarre le traçage avec le format avancé.

Cet attribut peut être lu par le client XMS .NET géré.

XMSDotnetTraceFilePath=nom_fichier

Si aucune valeur n'est définie pour l'attribut **XMSDotnetTraceFilePath** ou si cet attribut est présent mais qu'il contient une chaîne vide, le fichier de trace de XMS .NET est placé dans le répertoire de travail. Pour stocker le fichier de trace dans un répertoire nommé, indiquez le nom du répertoire dans le fichier **XMSDotnetTraceFilePath**, par exemple **XMSDotnetTraceFilePath="c:\somepath"**.

Cet attribut peut être lu par le client XMS .NET géré.

XMSDotnetTraceSpecification =ComponentName=type=état

L'attribut **XMSDotnetTraceSpecification** spécifie le nom de la classe que vous souhaitez tracer et le type de trace dont vous avez besoin pour XMS .NET:

- *ComponentName* est le nom de la classe que vous souhaitez tracer. Vous pouvez utiliser un caractère générique * dans ce nom. Par exemple, **=all=enabled* spécifie que vous voulez tracer toutes les classes, et *IBM.XMS.impl.*=all=enabled* que vous voulez tracer uniquement l'API.
- *type* peut être l'un des types de trace suivants: all, debug, event, EntryExit.
- *state* peut être activé ou désactivé.

Vous pouvez chaîner plusieurs éléments de trace ensemble en utilisant le délimiteur ":" (deux-points).

Cet attribut peut être lu par le client XMS .NET géré.

XMSDotnetTraceFileSize=taille

L'attribut **XMSDotnetTraceFileSize** indique la taille maximale du fichier de trace qui doit être généré pour XMS .NET. La valeur maximale par défaut est de 20 Mo, qui est spécifiée sous la forme **XMSDotnetTraceFileSize=20**.

Cet attribut peut être lu par le client XMS .NET géré.

XMSDotnetTraceFileNumber=numéro

L'attribut **XMSDotnetTraceFileNumber** indique le nombre de fichiers de trace à conserver pour XMS .NET. La valeur par défaut est 4 (un fichier actif et trois fichiers archive). Le nombre minimal autorisé est 2.

Cet attribut peut être lu par le client XMS .NET géré.

Tâches associées

[Traçage des applications IBM MQ .NET avec mqclient.ini](#)

[Traçage des applications XMS .NET avec mqclient.ini](#)

Multi

Fichier de configuration de trace d'activité, mqat.ini

Le fichier de configuration de trace d'activité, `mqat.ini`, est utilisé pour configurer le comportement de trace d'activité. Ce fichier est utilisé pour définir le niveau et la fréquence de génération de rapports sur les données de trace d'activité. Le fichier permet également de définir des règles pour activer et désactiver la trace d'activité en fonction du nom d'une application.

Le fichier `mqat.ini` suit le même format de paire de clé de section et de valeur de paramètre que les fichiers `mqs.ini` et `qm.ini`. Le fichier se compose d'une seule section, `AllActivityTrace`, qui est utilisée pour configurer le niveau et la fréquence des données de trace d'activité de rapport par défaut pour toutes les traces d'activité. Le fichier peut également contenir plusieurs sections `ApplicationTrace`. Chacune de ces sections définit une règle pour le comportement de trace d'une ou de plusieurs connexions, en fonction du nom d'application des connexions à la règle. Pour plus d'informations, voir [Trace d'activité d'application](#) et [Configuration du comportement de trace d'activité à l'aide de mqat.ini](#).

Le gestionnaire de files d'attente applique un certain nombre de règles pour déterminer les paramètres de strophes à utiliser pour une connexion. Vous pouvez éventuellement remplacer les paramètres de niveau et de fréquence de trace globale dans la section `AllActivityTrace` pour les connexions qui correspondent à une section `ApplicationTrace`. Pour plus d'informations, voir [Configuration du comportement de la trace d'activité à l'aide de mqat.ini](#).

emplacements des répertoires

IBM i **Linux** **AIX** Sur les systèmes AIX and Linux et IBM i, `mqat.ini` se trouve dans le répertoire de données du gestionnaire de files d'attente, qui est le même emplacement que le fichier `qm.ini`.

Windows Sur les systèmes Windows, `mqat.ini` se trouve dans le répertoire de données du gestionnaire de files d'attente `C:\Program Files\IBM\WebSphere MQ\mqgrs\queue_manager_name`. Les utilisateurs exécutant des applications à tracer doivent disposer des droits permettant de lire ce fichier.

Multi **Section de trace AllActivity du fichier mqat.ini**

La section de trace `AllActivity` du fichier de configuration `mqat.ini` spécifie les paramètres utilisés pour configurer les niveaux de trace d'un gestionnaire de files d'attente.

Une seule section `AllActivityTrace` définit les paramètres de la trace d'activité appliquée à toutes les connexions IBM MQ, sauf si elle est remplacée.

Les valeurs individuelles de la strophe de trace `AllActivity` peuvent être remplacées par des informations plus spécifiques dans une strophe `ApplicationTrace`.

Si plusieurs sections de trace `AllActivity` sont spécifiées, les valeurs de la dernière section sont utilisées. Les paramètres manquants dans la trace `AllActivity` choisie prennent les valeurs par défaut. Les paramètres et les valeurs des sections de trace `AllActivity` précédentes sont ignorés.

ActivityInterval

Intervalle de temps, en secondes, entre les messages de trace. La trace de l'activité n'utilise pas d'unité d'exécution de type minuteur ; par conséquent, le message trace n'est pas écrit dès que l'intervalle est écoulé, mais lorsque la première opération MQI est exécutée une fois l'intervalle de temps écoulé. Si la valeur est 0, le message trace est écrit lorsque la connexion est arrêtée (ou lorsque le nombre d'activités est atteint). La valeur par défaut est 1.

ActivityCount

Nombre d'opérations MQI entre les messages de trace. Si la valeur est 0, le message trace est écrit lorsque la connexion est arrêtée (ou lorsque l'intervalle de l'activité est écoulé). La valeur par défaut est 100.

TraceLevel

Quantité de détails de paramètre tracée pour chaque opération. La description des opérations individuelles indique les paramètres qui sont inclus pour chaque niveau de trace. Définissez la valeur sur `LOW`, `MEDIUM` ou `HIGH`. La valeur par défaut est `MEDIUM`.

TraceMessageData

Quantité de données de message tracée en octets pour les opérations `MQGET`, `MQPUT`, `MQPUT1` et de rappel. La valeur par défaut est 0.

StopOnGetTraceMsg

Peut être défini sur `ON` ou `OFF`. La valeur par défaut est `ON`.

SubscriptionDelivery

Peut être défini sur `BATCHED` ou `IMMEDIATE`. Détermine si les paramètres **ActivityInterval** et **ActivityCount** doivent être utilisés lorsqu'un ou plusieurs abonnements de trace d'activité sont présents. La définition de ce paramètre sur `IMMEDIATE` entraîne le remplacement des valeurs **ActivityInterval** et **ActivityCount** par les valeurs effectives 1 lorsque les données de trace ont un abonnement correspondant. Les enregistrements de trace de l'activité ne sont pas regroupés avec d'autres enregistrements de la même connexion, mais sont distribués à l'abonnement

immédiatement. Le paramètre IMMEDIATE augmente le temps système de la collecte des données de trace d'activité. La valeur par défaut est BATCHED.

Tâches associées

[Configuration du comportement de trace d'activité à l'aide de mqat.ini](#)

Multi Section ApplicationTrace du fichier mqat.ini

Le fichier de configuration mqat.ini peut contenir plusieurs sections ApplicationTrace. Chacune de ces sections définit une règle pour le comportement de trace d'une ou de plusieurs connexions, en fonction du nom d'application des connexions à la règle.

Vous pouvez définir les valeurs suivantes pour la section ApplicationTrace :

Fonction de trace

Commutateur de trace d'activité qui peut être défini sur ON ou OFF. Le paramètre **Trace** est un paramètre obligatoire sans valeur par défaut. Il peut être utilisé dans la section spécifique à l'application pour déterminer si la trace d'activité est active pour la portée de la section d'application en cours. Notez que cette valeur remplace les paramètres **ACTVTRC** et **ACTVCONO** du gestionnaire de files d'attente.

AppName

Le paramètre **AppName** est spécifié sous la forme d'une chaîne de caractères et est un paramètre obligatoire sans valeur par défaut. Cette valeur est utilisée pour déterminer les applications auxquelles la strophe ApplicationTrace s'applique. Il est mis en correspondance avec la valeur **AppName** de la structure de contexte d'exit d'API (qui est équivalente à MQMD.PutAppName). Le contenu de la valeur **AppName** varie en fonction de l'environnement d'application.

Sur Multiplatforms, uniquement la partie nom de fichier de MQAXC.AppName est mis en correspondance avec la valeur de la strophe. Les caractères situés à gauche du séparateur de chemin le plus à droite sont ignorés lors de la comparaison.

Un seul caractère générique (*) peut être utilisé à la fin de la valeur **AppName** pour correspondre à n'importe quel nombre de caractères après ce point. Si la valeur **AppName** est définie sur un caractère générique unique (*), la valeur **AppName** correspond à toutes les applications.

IBM i ApplFunction

Le paramètre **AppFunction** est spécifié sous la forme d'une chaîne de caractères. La valeur par défaut est *, La valeur de ce paramètre est utilisée pour déterminer à quelle application s'applique la section ApplicationTrace et à laquelle s'applique la valeur **AppName**.

La strophe est facultative et n'est valide que pour les gestionnaires de files d'attente IBM i. Un seul caractère générique (*) peut être utilisé à la fin de la valeur **AppName** pour correspondre à n'importe quel nombre de caractères. Par exemple, une section ApplicationTrace spécifiant **AppName** = * et **AppFunction** = AMQSPUTO s'applique à tous les appels du programme AMQSPUTO à partir de n'importe quel travail.

AppClass

Le paramètre **AppClass** définit la classe d'une application et peut être défini sur les valeurs suivantes:

- UTILISATEUR
- MCA
- ALL (valeur par défaut)

Pour une explication de la façon dont les valeurs **AppType** correspondent aux connexions IBM MQ, voir le [Tableau 3](#) dans [Configuration du comportement de la trace d'activité à l'aide de mqat.ini](#).

En option, les paramètres de niveau et de fréquence de trace globale sous la section de trace AllActivity peuvent être remplacés pour les connexions correspondant à une section ApplicationTrace.

Les paramètres suivants peuvent être définis dans une section ApplicationTrace. S'ils ne sont pas définis, la valeur est héritée de la section de trace [AllActivity](#) :

ActivityInterval

Intervalle de temps, en secondes, entre les messages de trace. La trace de l'activité n'utilise pas d'unité d'exécution de type minuteur ; par conséquent, le message trace n'est pas écrit dès que l'intervalle est écoulé, mais lorsque la première opération MQI est exécutée une fois l'intervalle de temps écoulé. Si la valeur est 0, le message trace est écrit lorsque la connexion est arrêtée (ou lorsque le nombre d'activités est atteint). La valeur par défaut est 1.

ActivityCount

Nombre d'opérations MQI entre les messages de trace. Si la valeur est 0, le message trace est écrit lorsque la connexion est arrêtée (ou lorsque l'intervalle de l'activité est écoulé). La valeur par défaut est 100.

TraceLevel

Quantité de détails de paramètre tracée pour chaque opération. La description des opérations individuelles indique les paramètres qui sont inclus pour chaque niveau de trace. Définissez la valeur sur LOW, MEDIUM ou HIGH. La valeur par défaut est MEDIUM.

TraceMessageData

Quantité de données de message tracée en octets pour les opérations MQGET, MQPUT, MQPUT1 et de rappel. La valeur par défaut est 0.

StopOnGetTraceMsg

Peut être défini sur ON ou OFF. La valeur par défaut est ON.

Tâches associées

[Configuration du comportement de trace d'activité à l'aide de mqat.ini](#)

Configuration de la file d'attente répartie



Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ, y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Avant de commencer

Avant de lire cette section, il est utile de comprendre les canaux, les files d'attente et les autres concepts introduits dans la rubrique [Mise en file d'attente répartie et clusters](#).

Si vous devez connecter deux gestionnaires de files d'attente qui se trouvent sur des réseaux physiques différents ou communiquer via un pare-feu, l'utilisation d' IBM MQ Internet Pass-Thru peut simplifier la configuration. Pour plus d'informations, voir [IBM MQ Internet Pass-Thru](#).

Procédure

- Utilisez les informations des sous-rubriques suivantes pour connecter vos applications à l'aide de la mise en file d'attente répartie:
 - [«Techniques de mise en file d'attente répartie IBM MQ»](#), à la page 213
 - [«Introduction à la gestion de files d'attente réparties»](#), à la page 235
 - [«Comment envoyer un message à un autre gestionnaire de files d'attente»](#), à la page 238
 - [«Canaux de déclenchement»](#), à la page 260
 - [«Sécurité des messages»](#), à la page 258
 -  [«Surveillance et contrôle des canaux sur AIX, Linux, and Windows»](#), à la page 268
 -  [«Surveillance et contrôle des canaux sur IBM i»](#), à la page 293

Concepts associés

[«Setting up IBM MQ for z/OS»](#), à la page 935

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Tâches associées

[«Configuration des connexions entre le client et le serveur», à la page 17](#)

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

[«Configuration d'un cluster de gestionnaires de files d'attente», à la page 316](#)

Les clusters fournissent un mécanisme d'interconnexion des gestionnaires de files d'attente qui simplifie à la fois la configuration initiale et la gestion en cours. Vous pouvez définir des composants de cluster et créer et gérer des clusters.

[«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms», à la page 96](#)

Vous pouvez modifier le comportement d' IBM MQ ou d'un gestionnaire de files d'attente individuel en fonction des besoins de votre installation en éditant les informations dans les fichiers de configuration (.ini). Vous pouvez également modifier les options de configuration pour IBM MQ MQI clients.

[«Configuring queue managers on z/OS», à la page 930](#)

Use these instructions to configure queue managers on IBM MQ for z/OS.

[«Setting up communications with other queue managers on z/OS», à la page 1009](#)

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Techniques de mise en file d'attente répartie IBM MQ

Les sous-rubriques de cette section décrivent les techniques utilisées lors de la planification des canaux. Ces sous-rubriques décrivent les techniques permettant de planifier la connexion de vos gestionnaires de files d'attente et de gérer le flux de messages entre vos applications.

Pour obtenir des exemples de planification de canal de message, voir:

- ▶ **ALW** [Exemple de planification de canal de message pour AIX, Linux, and Windows](#)
- ▶ **IBM i** [Exemple de planification de canal de message pour IBM i](#)
- ▶ **z/OS** [Exemple de planification de canal de message pour z/OS](#)
- ▶ **z/OS** [Exemple de planification de canal de message pour z/OS à l'aide de groupes de partage de files d'attente](#)

Concepts associés

[Canaux](#)

[Présentation de la mise en file d'attente de messages](#)

[Mise en file d'attente répartie et clusters](#)

Tâches associées

[«Configuration de la file d'attente répartie», à la page 212](#)


Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ , y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Référence associée

[Exemple d'informations de configuration](#)

contrôle du flux de messages

Le contrôle du flux de messages est une tâche qui implique la configuration et la maintenance des routes de messages entre les gestionnaires de files d'attente. Il est important pour les routes qui passent par de nombreux gestionnaires de files d'attente. Cette section explique comment utiliser les files d'attente, les définitions de file d'attente d'alias et les canaux de messages sur votre système pour contrôler le flux de messages.

Vous contrôlez le flux de messages à l'aide d'un certain nombre de techniques qui ont été introduites dans «Configuration de la file d'attente répartie», à la page 212. Si votre gestionnaire de files d'attente se trouve dans un cluster, le flux de messages est contrôlé à l'aide de différentes techniques, comme décrit dans «contrôle du flux de messages», à la page 213.  Si vos gestionnaires de files d'attente se trouvent dans un groupe de partage de files d'attente et que la mise en file d'attente intra-groupe est activée, le flux de messages peut être contrôlé par des agents de mise en file d'attente intra-groupe. Ces agents sont décrits dans [Mise en file d'attente intra-groupe](#).

Vous pouvez utiliser les objets suivants pour obtenir le contrôle du flux de messages:

- Files d'attente de transmission
- Canaux de transmission de messages
- Définition de file d'attente éloignée
- Définition d'alias de gestionnaire de files d'attente
- Définition d'alias de file d'attente de réponse

Les objets de gestionnaire de files d'attente et de file d'attente sont décrits dans [Types d'objet](#). Les canaux de transmission de messages sont décrits dans [Composants de la mise en file d'attente répartie](#). Les techniques suivantes utilisent ces objets pour créer des flux de messages dans votre système:

- Insertion de messages dans des files d'attente éloignées
- Routage au moyen de files d'attente de transmission particulières
- Réception de messages
- Transmission de messages via votre système
- Séparation des flux de messages
- Basculement d'un flux de messages vers une autre destination
- Résolution du nom de la file d'attente de réponse en nom d'alias

Remarque

Tous les concepts décrits dans cette section sont pertinents pour tous les noeuds d'un réseau et incluent l'envoi et la réception des extrémités des canaux de message. Pour cette raison, un seul noeud est illustré dans la plupart des exemples. L'exception est que l'exemple nécessite une coopération explicite de la part de l'administrateur à l'autre extrémité d'un canal de transmission de messages.

Avant de passer aux techniques individuelles, il est utile de récapituler les concepts de résolution de nom et les trois façons d'utiliser les définitions de file d'attente éloignée. Voir [Mise en file d'attente répartie et clusters](#).

Concepts associés

«Noms de file d'attente dans l'en-tête de transmission», à la page 214

Les noms de file d'attente de destination sont associés au message dans l'en-tête de transmission jusqu'à ce que la file d'attente de destination soit atteinte.

«Comment créer des alias de gestionnaire de files d'attente et de réponse», à la page 215

Cette rubrique explique les trois façons de créer une définition de file d'attente éloignée.

Noms de file d'attente dans l'en-tête de transmission

Les noms de file d'attente de destination sont associés au message dans l'en-tête de transmission jusqu'à ce que la file d'attente de destination soit atteinte.

Le nom de file d'attente utilisé par l'application, le nom de file d'attente logique, est résolu par le gestionnaire de files d'attente en nom de file d'attente de destination. En d'autres termes, le nom de la file d'attente physique. Ce nom de file d'attente de destination se déplace avec le message dans une zone de données distincte, l'en-tête de transmission, jusqu'à ce que la file d'attente de destination soit atteinte. L'en-tête de transmission est ensuite retiré.

Vous modifiez la partie gestionnaire de files d'attente de ce nom de file d'attente lorsque vous créez des classes de service parallèles. N'oubliez pas de renvoyer le nom du gestionnaire de files d'attente au nom d'origine lorsque la fin de la déviation de la classe de service a été atteinte.

Comment créer des alias de gestionnaire de files d'attente et de réponse

Cette rubrique explique les trois façons de créer une définition de file d'attente éloignée.

L'objet de définition de file d'attente éloignée est utilisé de trois manières différentes. [Tableau 17](#), à la page 215 explique comment définir chacune des trois manières suivantes:

- Utilisation d'une définition de file d'attente éloignée pour redéfinir un nom de file d'attente locale.

L'application fournit uniquement le nom de la file d'attente lors de l'ouverture d'une file d'attente et ce nom correspond au nom de la définition de file d'attente éloignée.

La définition de file d'attente éloignée contient les noms de la file d'attente cible et du gestionnaire de files d'attente. La définition peut éventuellement contenir le nom de la file d'attente de transmission à utiliser. Si aucun nom de file d'attente de transmission n'est fourni, le gestionnaire de files d'attente utilise le nom du gestionnaire de files d'attente, extrait de la définition de file d'attente éloignée, pour le nom de la file d'attente de transmission. Si une file d'attente de transmission de ce nom n'est pas définie, mais qu'une file d'attente de transmission par défaut est définie, la file d'attente de transmission par défaut est utilisée.

- Utilisation d'une définition de file d'attente éloignée pour redéfinir un nom de gestionnaire de files d'attente.

L'application ou le programme de canal fournit un nom de file d'attente avec le nom du gestionnaire de files d'attente éloignées lors de l'ouverture de la file d'attente.

Si vous avez fourni une définition de file d'attente éloignée portant le même nom que le nom du gestionnaire de files d'attente et que vous avez laissé le nom de la file d'attente dans la définition à blanc, le gestionnaire de files d'attente remplace le nom du gestionnaire de files d'attente dans l'appel ouvert par le nom du gestionnaire de files d'attente dans la définition.

En outre, la définition peut contenir le nom de la file d'attente de transmission à utiliser. Si aucun nom de file d'attente de transmission n'est fourni, le gestionnaire de files d'attente prend le nom du gestionnaire de files d'attente, extrait de la définition de file d'attente éloignée, pour le nom de la file d'attente de transmission. Si une file d'attente de transmission de ce nom n'est pas définie, mais qu'une file d'attente de transmission par défaut est définie, la file d'attente de transmission par défaut est utilisée.

- Utilisation d'une définition de file d'attente éloignée pour redéfinir un nom de file d'attente de réponse.

Chaque fois qu'une application insère un message dans une file d'attente, elle peut fournir le nom d'une file d'attente de réponse pour les messages de réponse, mais avec le nom du gestionnaire de files d'attente vide.

Si vous fournissez une définition de file d'attente éloignée portant le même nom que la file d'attente de réponse, le gestionnaire de files d'attente local remplace le nom de file d'attente de réponse par le nom de file d'attente de votre définition.

Vous pouvez indiquer un nom de gestionnaire de files d'attente dans la définition, mais pas un nom de file d'attente de transmission.

<i>Tableau 17. Trois façons d'utiliser l'objet de définition de file d'attente éloignée</i>			
Utilisation	Nom gest. de files	Nom de la file d'attente	Nom de la file d'attente de transmission
1. Définition de la file d'attente éloignée (lors de l'appel OPEN)			
Fourni dans l'appel	vide ou gestionnaire de files d'attente local	(*) obligatoire	non applicable

Tableau 17. Trois façons d'utiliser l'objet de définition de file d'attente éloignée (suite)

Utilisation	Nom gest. de files	Nom de la file d'attente	Nom de la file d'attente de transmission
Fourni dans la définition	obligatoire	obligatoire	facultatif
2. Alias du gestionnaire de files d'attente (sur appel OPEN)			
Fourni dans l'appel	(*) requis et non local QM	obligatoire	non applicable
Fourni dans la définition	obligatoire	blanc	facultatif
3. Alias de file d'attente de réponse (lors de l'appel PUT)			
Fourni dans l'appel	blanc	(*) obligatoire	non applicable
Fourni dans la définition	facultatif	facultatif	blanc

Remarque : (*) signifie que ce nom est le nom de l'objet de définition

Pour une description formelle, voir [Résolution de nom de file d'attente](#).

Insertion de messages dans des files d'attente éloignées

Vous pouvez utiliser des objets de définition de file d'attente éloignée pour résoudre un nom de file d'attente en file d'attente de transmission dans un gestionnaire de files d'attente adjacent.

Dans un environnement de mise en file d'attente répartie, une file d'attente de transmission et un canal sont le point focal de tous les messages vers un emplacement, que les messages proviennent d'applications de votre système local ou arrivent via des canaux d'un système adjacent. La [Figure 6](#), à la [page 217](#) montre une application plaçant des messages dans une file d'attente logique nommée 'QA_norm'. La résolution de nom utilise la définition de file d'attente éloignée 'QA_norm' pour sélectionner la file d'attente de transmission QMB. Il ajoute ensuite un en-tête de transmission aux messages indiquant 'QA_norm at QMB'.

Les messages arrivant du système adjacent sur 'Channel_back' ont un en-tête de transmission avec le nom de file d'attente physique 'QA_norm at QMB', par exemple. Ces messages sont placés inchangés dans la file d'attente de transmission QMB.

Le canal déplace les messages vers un gestionnaire de files d'attente adjacent.

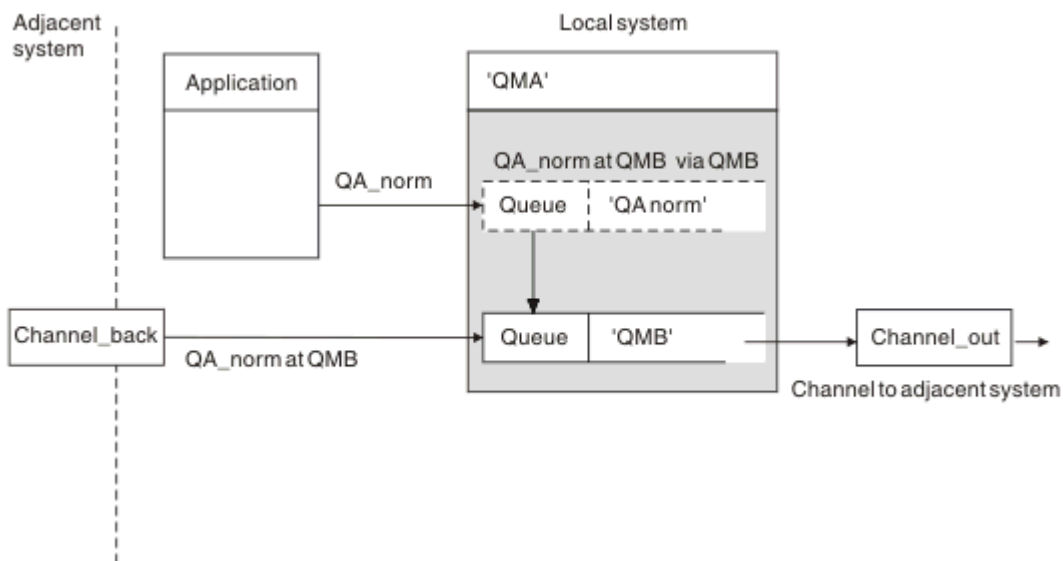


Figure 6. Une définition de file d'attente éloignée est utilisée pour résoudre un nom de file d'attente en file d'attente de transmission vers un gestionnaire de files d'attente adjacent.

Si vous êtes l'administrateur système IBM MQ , vous devez:

- Définir le canal de transmission de messages à partir du système adjacent
- Définir le canal de transmission de messages sur le système adjacent
- Création de la file d'attente de transmission QMB
- Définissez l'objet de file d'attente éloignée 'QA_norm' pour résoudre le nom de file d'attente utilisé par les applications en nom de file d'attente de destination, en nom de gestionnaire de files d'attente de destination et en nom de file d'attente de transmission

Dans un environnement en cluster, il suffit de définir un canal récepteur de cluster au niveau du gestionnaire de files d'attente local. Il n'est pas nécessaire de définir une file d'attente de transmission ou un objet de file d'attente éloignée. Voir [Clusters](#).

En savoir plus sur la résolution de nom

La définition de file d'attente éloignée a pour effet de définir un nom de file d'attente de destination physique et un nom de gestionnaire de files d'attente. Ces noms sont insérés dans les en-têtes de transmission des messages.

Ce type de résolution de nom a déjà été effectué par le gestionnaire de files d'attente d'origine pour les messages entrants provenant d'un système adjacent. Par conséquent, l'en-tête de transmission indique le nom de la file d'attente de destination physique et le nom du gestionnaire de files d'attente. Ces messages ne sont pas affectés par les définitions de file d'attente éloignée.

Référence associée

[Résolution de nom de file d'attente](#)

Choix de la file d'attente de transmission

Vous pouvez utiliser une définition de file d'attente éloignée pour permettre à une autre file d'attente de transmission d'envoyer des messages au même gestionnaire de files d'attente adjacent.

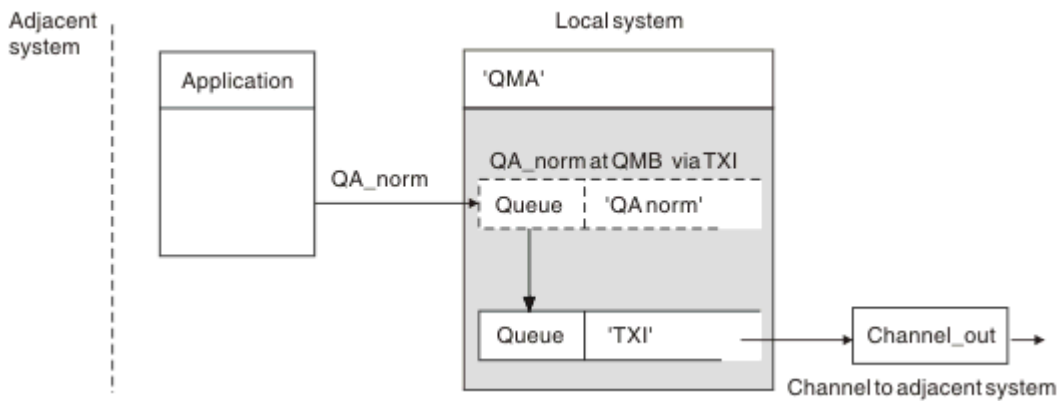


Figure 7. La définition de file d'attente éloignée permet d'utiliser une autre file d'attente de transmission

Dans un environnement de mise en file d'attente répartie, lorsque vous devez modifier un flux de messages d'un canal à un autre, utilisez la même configuration système que celle illustrée dans la [Figure 6](#), à la page 217 dans la «Insertion de messages dans des files d'attente éloignées», à la page 216. La [Figure 7](#), à la page 218 de cette rubrique explique comment utiliser la définition de file d'attente éloignée pour envoyer des messages via une file d'attente de transmission différente, et donc via un canal différent, vers le même gestionnaire de files d'attente adjacent.

Pour la configuration illustrée dans la [Figure 7](#), à la page 218, vous devez fournir l'objet de file d'attente éloignée 'QA_norm' et la file d'attente de transmission 'TXI'. Vous devez indiquer 'QA_norm' pour choisir la file d'attente 'QA_norm' sur le gestionnaire de files d'attente éloignées, la file d'attente de transmission 'TXI' et le gestionnaire de files d'attente 'QMB_priority'. Indiquez 'TXI' dans la définition du canal adjacent au système.

Les messages sont placés dans la file d'attente de transmission 'TXI' avec un en-tête de transmission contenant 'QA_norm at QMB_priority' et sont envoyés via le canal au système adjacent.

Le channel_back a été supprimé de cette illustration car il aurait besoin d'un alias de gestionnaire de files d'attente.

Dans un environnement de mise en cluster, il n'est pas nécessaire de définir une file d'attente de transmission ou une définition de file d'attente éloignée. Pour plus d'informations, voir «[Définition de files d'attente de cluster](#)», à la page 317.

Réception de messages

Vous pouvez configurer le gestionnaire de files d'attente pour qu'il reçoive des messages d'autres gestionnaires de files d'attente. Vous devez vous assurer que la résolution de nom non intentionnelle ne se produit pas.

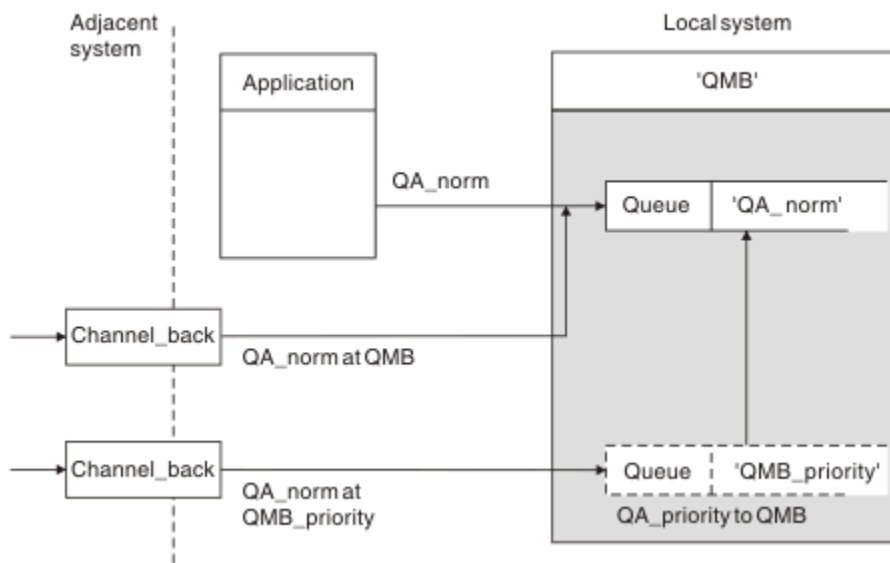


Figure 8. Réception directe des messages et résolution du nom du gestionnaire de files d'attente alias

En plus d'organiser l'envoi des messages, l'administrateur système doit également organiser la réception des messages provenant des gestionnaires de files d'attente adjacents. Les messages reçus contiennent le nom physique du gestionnaire de files d'attente de destination et de la file d'attente dans l'en-tête de transmission. Ils sont traités de la même manière que les messages provenant d'une application locale qui spécifie à la fois le nom du gestionnaire de files d'attente et le nom de la file d'attente. En raison de ce traitement, vous devez vous assurer que les messages entrant dans votre système ne font pas l'objet d'une résolution de nom non intentionnelle. Pour ce scénario, voir [Figure 8](#), à la page 219 .

Pour cette configuration, vous devez préparer:

- Canaux de transmission de messages destinés à recevoir des messages des gestionnaires de files d'attente adjacents
- Une définition d'alias de gestionnaire de files d'attente pour résoudre un flux de messages entrant, 'QMB_priority', en nom de gestionnaire de files d'attente local, 'QMB'
- La file d'attente locale, 'QA_norm', si elle n'existe pas

Réception des noms de gestionnaire de files d'attente alias

L'utilisation de la définition d'alias de gestionnaire de files d'attente dans cette illustration n'a pas sélectionné de gestionnaire de files d'attente de destination différent. Les messages transitant par ce gestionnaire de files d'attente local et adressés à 'QMB_priority' sont destinés au gestionnaire de files d'attente 'QMB'. Le nom du gestionnaire de files d'attente alias est utilisé pour créer le flux de messages distinct.

Transmission de messages via votre système

Vous pouvez transmettre des messages via votre système de trois manières: en utilisant le nom d'emplacement, en utilisant un alias pour le gestionnaire de files d'attente ou en sélectionnant une file d'attente de transmission.

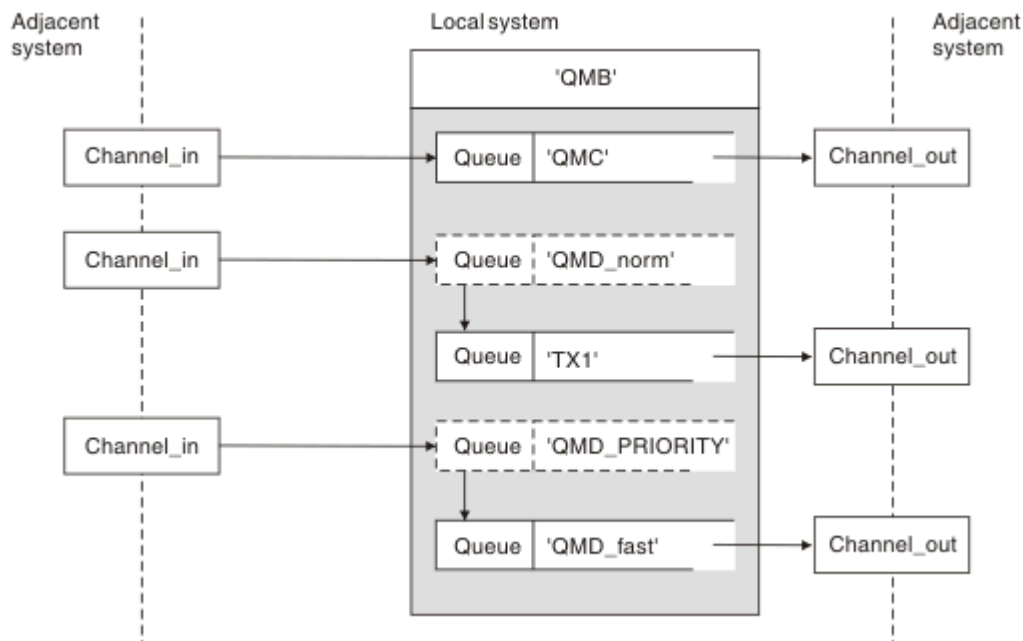


Figure 9. Trois méthodes de transmission de messages via votre système

La technique présentée dans la [Figure 8](#), à la page 219 dans la «Réception de messages», à la page 218 montre comment un flux d'alias est capturé. La [Figure 9](#), à la page 220 illustre la façon dont les réseaux sont créés en regroupant les techniques décrites précédemment.

La configuration montre un canal qui distribue trois messages avec des destinations différentes:

1. QB à QMC
2. QB à QMD_norm
3. QB à QMD_PRIORITY

Vous devez transmettre le premier flux de messages à votre système tel quel. Vous devez transmettre le deuxième flux de messages via une file d'attente de transmission et un canal différents. Pour le deuxième flux de messages, vous devez également résoudre les messages pour le nom de gestionnaire de files d'attente alias QMD_norm dans le gestionnaire de files d'attente QMD. Le troisième flux de messages choisit une file d'attente de transmission différente sans autre modification.

Dans un environnement de mise en cluster, les messages sont transmis via une file d'attente de transmission de cluster. Normalement, une file d'attente de transmission unique, SYSTEM.CLUSTER.TRANSMIT.QUEUE, transfère tous les messages à tous les gestionnaires de files d'attente de tous les clusters dont le gestionnaire de files d'attente est membre ; voir [Cluster de gestionnaires de files d'attente](#). Vous pouvez définir des files d'attente de transmission distinctes pour tout ou partie des gestionnaires de files d'attente dans les clusters dont le gestionnaire de files d'attente est membre.

Les méthodes suivantes décrivent les techniques applicables à un environnement de mise en file d'attente répartie.

Utilisez ces méthodes

Pour ces configurations, vous devez préparer les éléments suivants:

- Définitions de canal d'entrée
- Définitions de canal de sortie
- Files d'attente de transmission:
 - QMC

- TX1
- QMD_fast
- Définitions d'alias de gestionnaire de files d'attente:
 - QMD_norm avec QMD_norm vers QMD via TX1
 - QMD_PRIORITY avec QMD_PRIORITY vers QMD_PRIORITY via QMD_fast

Remarque : Aucun des flux de messages affichés dans l'exemple ne modifie la file d'attente de destination. Les alias de nom de gestionnaire de files d'attente permettent de séparer les flux de messages.

Méthode 1: Utiliser le nom de l'emplacement entrant

Vous allez recevoir des messages avec un en-tête de transmission contenant un autre nom d'emplacement, tel que QMC. La configuration la plus simple consiste à créer une file d'attente de transmission portant ce nom, QMC. Le canal qui dessert la file d'attente de transmission distribue le message inchangé à la destination suivante.

Méthode 2: Utiliser un alias pour le gestionnaire de files d'attente

La deuxième méthode consiste à utiliser la définition d'objet d'alias de gestionnaire de files d'attente, mais à spécifier un nouveau nom d'emplacement, QMD, et une file d'attente de transmission particulière, TX1. Cette action:

- Met fin au flux de messages d'alias configuré par l'alias de nom de gestionnaire de files d'attente QMD_norm, c'est-à-dire la classe de service nommée QMD_norm.
- Modifie les en-têtes de transmission de ces messages de QMD_norm à QMD.

Méthode 3: Sélection d'une file d'attente de transmission

La troisième méthode consiste à définir un objet d'alias de gestionnaire de files d'attente avec le même nom que l'emplacement de destination, QMD_PRIORITY. Utilisez la définition d'alias de gestionnaire de files d'attente pour sélectionner une file d'attente de transmission particulière, QMD_fast, et donc un autre canal. Les en-têtes de transmission de ces messages restent inchangés.

Séparation des flux de messages

Vous pouvez utiliser un alias de gestionnaire de files d'attente pour créer des flux de messages distincts afin d'envoyer des messages au même gestionnaire de files d'attente.

Raisons de la séparation des messages en différents flux de messages

Dans un environnement de mise en file d'attente répartie, la nécessité de séparer les messages envoyés au même gestionnaire de files d'attente dans des flux de messages différents peut se produire pour plusieurs raisons. Exemple :

- Vous devrez peut-être fournir un flux distinct pour les messages de grande taille, de taille moyenne et de petite taille. Ce besoin s'applique également dans un environnement de mise en cluster et, dans ce cas, vous pouvez créer des clusters qui se chevauchent. Il existe un certain nombre de raisons pour lesquelles vous pouvez le faire, par exemple:
 - Permettre à différentes organisations d'avoir leur propre administration.
 - Permettre l'administration séparée des applications indépendantes.
 - Pour créer une classe de service. Par exemple, vous pouvez avoir un cluster appelé STAFF qui est un sous-ensemble du cluster appelé ÉTUDE. Lorsque vous placez un message dans une file d'attente annoncée dans le cluster STAFF, un canal restreint est utilisé. Lorsque vous placez un message dans

une file d'attente annoncée dans le cluster ÉTUDE, un canal général ou un canal restreint peut être utilisé.

- Pour créer des environnements de test et de production.
- Il peut être nécessaire d'acheminer les messages entrants par des chemins différents à partir du chemin des messages générés en local.
- Votre installation peut avoir besoin de planifier le déplacement des messages à certains moments (par exemple, la nuit) et les messages doivent ensuite être stockés dans des files d'attente réservées jusqu'à leur planification.

Exemple de flux de messages

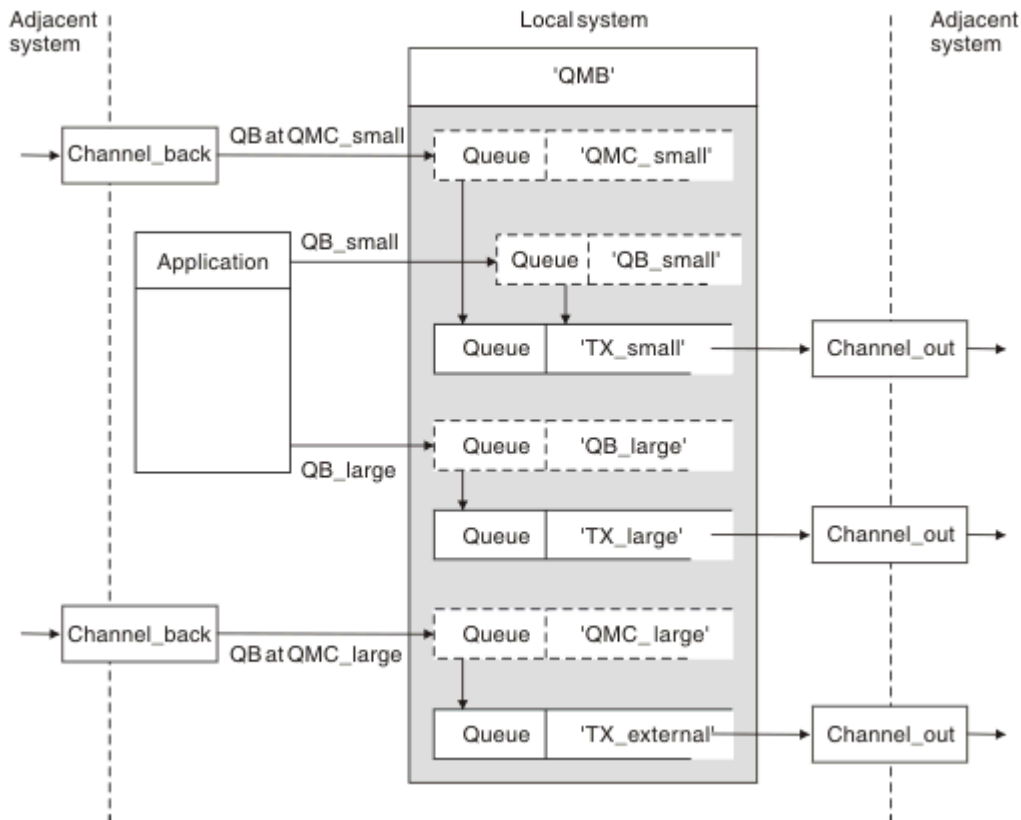


Figure 10. Séparation des flux de messages

Dans l'exemple illustré dans la [Figure 10](#), à la page 222, les deux flux entrants sont vers les noms de gestionnaire de files d'attente d'alias 'QMC_small' et 'QMC_large'. Vous fournissez ces flux avec une définition d'alias de gestionnaire de files d'attente pour capturer ces flux pour le gestionnaire de files d'attente local. Vous disposez d'une application qui traite deux files d'attente distantes et vous avez besoin que ces flux de messages soient séparés. Vous fournissez deux définitions de file d'attente éloignée qui indiquent le même emplacement, 'QMC', mais qui spécifient des files d'attente de transmission différentes. Cette définition maintient les flux séparés et aucun élément supplémentaire n'est nécessaire à l'extrémité éloignée car ils ont le même nom de gestionnaire de files d'attente de destination dans les en-têtes de transmission. Vous fournissez:

- Définitions de canal entrant
- Les deux définitions de file d'attente éloignée QB_small et QB_large
- Les deux définitions d'alias de gestionnaire de files d'attente QMC_small et QMC_large
- Les trois définitions de canal émetteur
- Trois files d'attente de transmission: TX_small, TX_large et TX_external

Coordination avec les systèmes adjacents

Lorsque vous utilisez un alias de gestionnaire de files d'attente pour créer un flux de messages distinct, vous devez coordonner cette activité avec l'administrateur système à l'extrémité éloignée du canal de transmission de messages pour vous assurer que l'alias de gestionnaire de files d'attente correspondant y est disponible.

Concentration des messages dans divers emplacements

Vous pouvez concentrer les messages destinés à différents emplacements sur un seul canal.

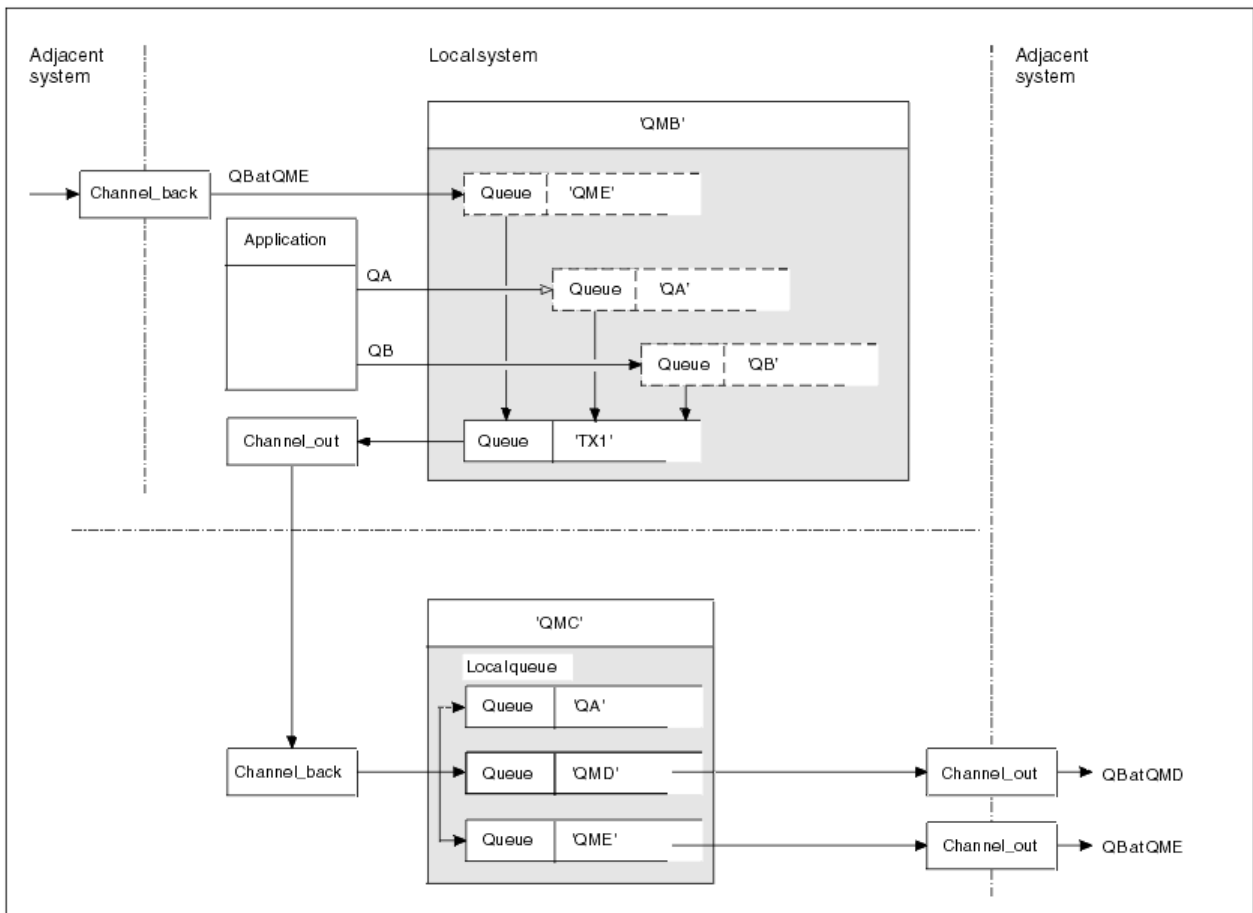


Figure 11. Combinaison de flux de messages sur un canal

La Figure 11, à la page 223 illustre une technique de mise en file d'attente répartie pour la concentration des messages destinés à différents emplacements sur un canal. Deux utilisations possibles seraient:

- Concentration du trafic des messages via une passerelle
- Utilisation d'autoroutes à large bande passante entre les noeuds

Dans cet exemple, les messages provenant de sources différentes, locales et adjacentes, et ayant des files d'attente de destination et des gestionnaires de files d'attente différents, sont transmis via la file d'attente de transmission 'TX1' au gestionnaire de files d'attente QMC. Le gestionnaire de files d'attente QMC distribue les messages en fonction des destinations. Une est définie sur une file d'attente de transmission 'QMD' pour la transmission vers le gestionnaire de files d'attente QMD. Un autre ensemble est défini sur une file d'attente de transmission 'QME' pour la transmission vers le gestionnaire de files d'attente QME. D'autres messages sont placés dans la file d'attente locale 'QA'.

Vous devez fournir:

- Définitions de canal
- File d'attente de transmission TX1
- Définitions de file d'attente éloignée:
 - QA avec 'QA at QMC through TX1'
 - QB avec 'QB à QMD via TX1'
- Définition d'alias de gestionnaire de files d'attente:
 - QME avec 'QME via TX1'

L'administrateur complémentaire qui configure QMC doit fournir:

- Définition de canal de réception avec le même nom de canal
- File d'attente de transmission QMD avec définition de canal émetteur associée
- File d'attente de transmission QME avec définition de canal d'envoi associée
- File d'attente locale de l'objet QA.

Détournement des flux de messages vers une autre destination

Vous pouvez redéfinir la destination de certains messages à l'aide d'alias de gestionnaire de files d'attente et de files d'attente de transmission.

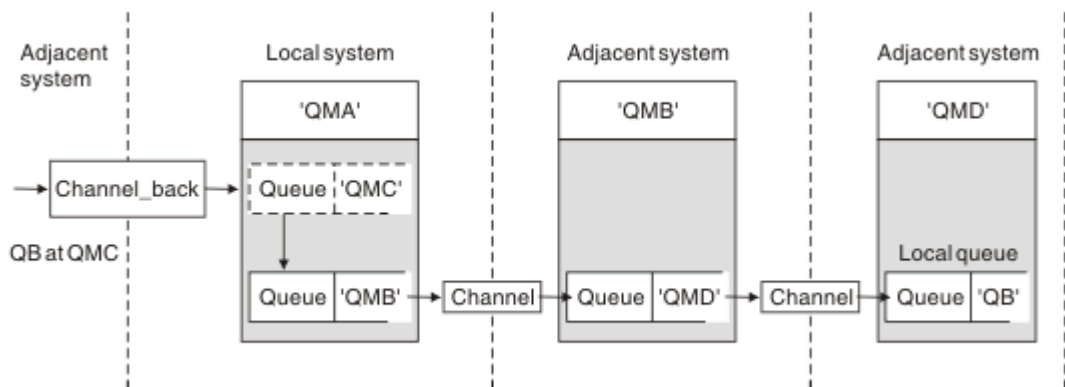


Figure 12. Détournement des flux de messages vers une autre destination

La Figure 12, à la page 224 montre comment redéfinir la destination de certains messages. Les messages entrants dans QMA sont destinés à 'QB at QMC'. Ils arrivent normalement à QMA et sont placés sur une file d'attente de transmission appelée QMC qui a fait partie d'un canal vers QMC. QMA doit dévier les messages vers QMD, mais il ne peut atteindre QMD que sur QMB. Cette méthode est utile lorsque vous devez déplacer un service d'un emplacement à un autre et permettre aux abonnés de continuer à envoyer des messages sur une base temporaire jusqu'à ce qu'ils aient été ajustés à la nouvelle adresse.

La méthode de réacheminement des messages entrants destinés à un gestionnaire de files d'attente donné vers un gestionnaire de files d'attente différent utilise:

- Un alias de gestionnaire de files d'attente pour remplacer le gestionnaire de files d'attente de destination par un autre gestionnaire de files d'attente et pour sélectionner une file d'attente de transmission sur le système adjacent
- Une file d'attente de transmission pour servir le gestionnaire de files d'attente adjacent
- Une file d'attente de transmission sur le gestionnaire de files d'attente adjacent pour le routage vers le gestionnaire de files d'attente de destination

Vous devez fournir:

- Définition channel_back
- Définition d'objet d'alias de gestionnaire de files d'attente QMC avec QB au niveau de QMD à QMB

- Définition channel_out
- File d'attente de transmission associée (QMB)

L'administrateur complémentaire qui configure QMB doit fournir:

- Définition channel_back correspondante
- File d'attente de transmission, QMD
- Définition de canal associée à QMD

Vous pouvez utiliser des alias dans un environnement de mise en cluster. Pour plus d'informations, voir «Alias de gestionnaire de files d'attente et clusters», à la page 418.

Envoi de messages à une liste de distribution

Sur Multiplatforms, vous pouvez utiliser un seul appel MQPUT pour qu'une application envoie un message à plusieurs destinations.

Dans IBM MQ on Multiplatforms, une application peut envoyer un message à plusieurs destinations avec un seul appel MQPUT. Vous pouvez le faire à la fois dans un environnement de mise en file d'attente répartie et dans un environnement de mise en cluster. Vous devez définir les destinations dans une liste de distribution, comme décrit dans [Listes de distribution](#).

Tous les gestionnaires de files d'attente ne prennent pas en charge les listes de distribution. Lorsqu'un agent MCA établit une connexion avec un partenaire, il détermine si le partenaire prend en charge les listes de distribution et définit un indicateur sur la file d'attente de transmission en conséquence. Si une application tente d'envoyer un message destiné à une liste de distribution mais que le partenaire ne prend pas en charge les listes de distribution, l'agent MCA émetteur intercepte le message et le place dans la file d'attente de transmission une fois pour chaque destination prévue.

Un agent MCA récepteur garantit que les messages envoyés à une liste de distribution sont reçus en toute sécurité sur toutes les destinations prévues. Si des destinations échouent, l'agent MCA détermine celles qui ont échoué. Il peut ensuite générer des rapports d'exception pour eux et essayer de leur envoyer à nouveau les messages.

File de réponses

Vous pouvez créer une boucle de traitement de file d'attente éloignée complète à l'aide d'une file d'attente de réponse.

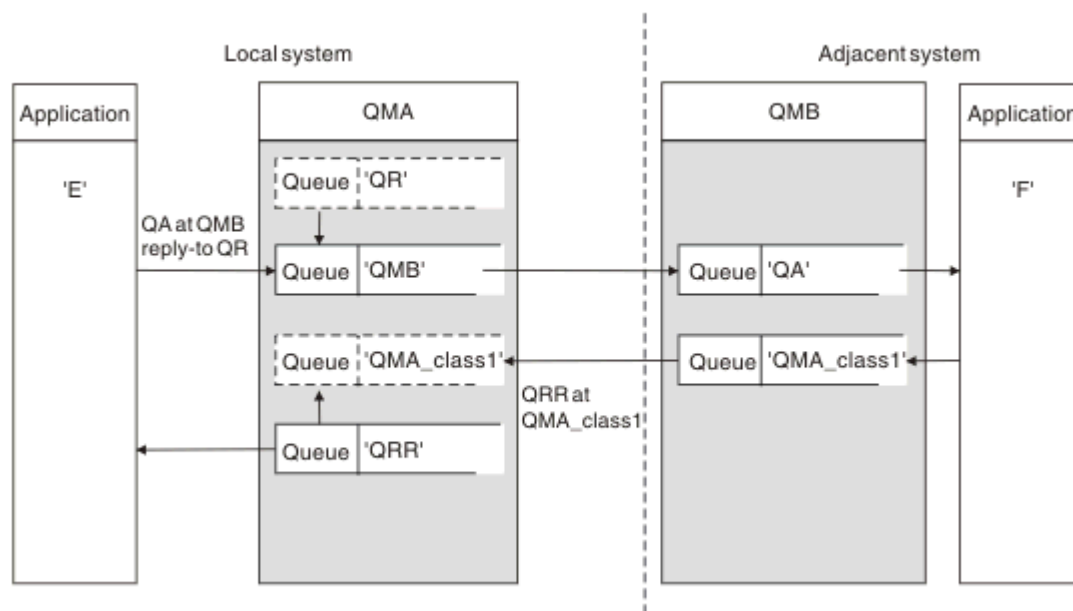


Figure 13. Substitution de nom de file d'attente de réponse lors de l'appel PUT

Une boucle de traitement de file d'attente éloignée complète utilisant une file d'attente de réponse est présentée dans la [Figure 13](#), à la [page 225](#). Cette boucle s'applique à la fois à un environnement de mise en file d'attente répartie et à un environnement de mise en cluster. Les détails sont présentés dans la [Tableau 21](#), à la [page 234](#).

L'application ouvre QA au niveau de QMB et place les messages dans cette file d'attente. Les messages reçoivent un nom de file d'attente de réponse QR, sans que le nom du gestionnaire de files d'attente soit spécifié. Le gestionnaire de files d'attente QMA recherche l'objet de file d'attente de réponse QR et en extrait le nom d'alias de QRR et le nom de gestionnaire de files d'attente QMA_class1. Ces noms sont placés dans les zones de réponse des messages.

Les messages de réponse des applications de QMB sont adressés à QRR à l'adresse QMA_class1. La définition de nom d'alias de gestionnaire de files d'attente QMA_class1 est utilisée par le gestionnaire de files d'attente pour transmettre les messages à lui-même et à la file d'attente QRR.

Ce scénario décrit la façon dont vous donnez aux applications la possibilité de choisir une classe de service pour les messages de réponse. La classe est implémentée par la file d'attente de transmission QMA_class1 à QMB, avec la définition d'alias de gestionnaire de files d'attente, QMA_class1 à QMA. De cette manière, vous pouvez modifier la file d'attente de réponse d'une application de sorte que les flux soient séparés sans impliquer l'application. L'application choisit toujours QR pour cette classe de service particulière. Vous avez la possibilité de modifier la classe de service avec la définition de file d'attente de réponse QR.

Vous devez créer:

- QR de définition de file d'attente de réponse
- Objet file d'attente de transmission QMB
- Définition channel_out
- Définition channel_back
- Définition d'alias de gestionnaire de files d'attente QMA_class1
- Objet file d'attente locale QRR, s'il n'existe pas

L'administrateur complémentaire du système adjacent doit créer:

- Définition de canal de réception
- Objet file d'attente de transmission QMA_class1
- Canal émetteur associé
- File d'attente locale de l'objet QA.

Vos programmes d'application utilisent:

- Nom de la file d'attente de réponse QR dans les appels d'insertion
- Nom de file d'attente QRR dans les appels d'extraction

De cette manière, vous pouvez modifier la classe de service si nécessaire, sans impliquer l'application. Vous modifiez l'alias de réponse 'QR' avec la file d'attente de transmission 'QMA_class1' et l'alias de gestionnaire de files d'attente 'QMA_class1'.

Si aucun objet alias de réponse n'est trouvé lorsque le message est inséré dans la file d'attente, le nom du gestionnaire de files d'attente local est inséré dans la zone de nom du gestionnaire de files d'attente de réponse vide. Le nom de la file d'attente de réponse reste inchangé.

Restriction de résolution de nom

Etant donné que la résolution de nom a été effectuée pour la file d'attente de réponse à 'QMA' lors de l'insertion du message d'origine, aucune autre résolution de nom n'est autorisée à 'QMB'. Le message est inséré avec le nom physique de la file d'attente de réponse par l'application de réponse.

Les applications doivent savoir que le nom qu'elles utilisent pour la file d'attente de réponse est différent du nom de la file d'attente réelle dans laquelle les messages de retour doivent être trouvés.

Par exemple, lorsque deux classes de service sont fournies pour l'utilisation d'applications avec les noms d'alias de file d'attente de réponse 'C1_alias' et 'C2_alias', les applications utilisent ces noms comme noms de file d'attente de réponse dans les appels d'insertion de message. Toutefois, les applications s'attendent à ce que des messages apparaissent dans les files d'attente 'C1' pour 'C1_alias' et 'C2' pour 'C2_alias'.

Toutefois, une application peut effectuer un appel d'interrogation sur la file d'attente alias de réponse pour vérifier elle-même le nom de la file d'attente réelle qu'elle doit utiliser pour obtenir les messages de réponse.

Concepts associés

«Comment créer des alias de gestionnaire de files d'attente et de réponse», à la page 215

Cette rubrique explique les trois façons de créer une définition de file d'attente éloignée.

«Exemple d'alias de file d'attente de réponse», à la page 227

Cet exemple illustre l'utilisation d'un alias de réponse pour sélectionner une route différente (file d'attente de transmission) pour les messages renvoyés. L'utilisation de cette fonction nécessite que le nom de la file d'attente de réponse soit modifié en coopération avec les applications.

«Fonctionnement de l'exemple», à la page 229

Une explication de l'exemple et de la façon dont le gestionnaire de files d'attente utilise l'alias de file d'attente de réponse.

«Présentation de l'alias de file d'attente de réponse», à la page 229

Une revue du processus à partir d'une application qui place un message dans une file d'attente éloignée dans la même application en supprimant le message de réponse de la file d'attente de réponse alias.

Exemple d'alias de file d'attente de réponse

Cet exemple illustre l'utilisation d'un alias de réponse pour sélectionner une route différente (file d'attente de transmission) pour les messages renvoyés. L'utilisation de cette fonction nécessite que le nom de la file d'attente de réponse soit modifié en coopération avec les applications.

Comme illustré dans la Figure 14, à la page 227, la route de retour doit être disponible pour les messages de réponse, y compris la file d'attente de transmission, le canal et l'alias du gestionnaire de files d'attente.

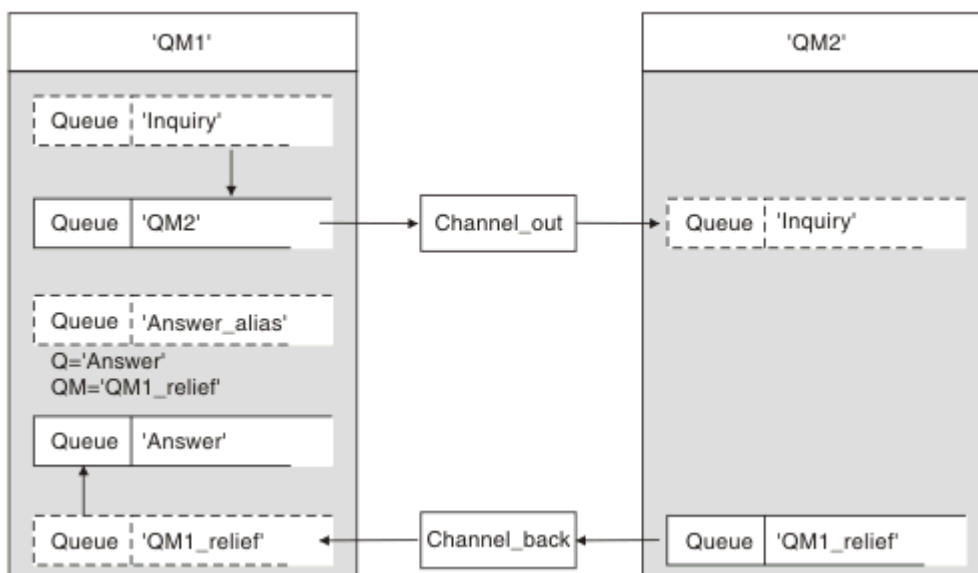


Figure 14. Exemple d'alias de file d'attente de réponse

Cet exemple concerne les applications de demandeur à l'adresse 'QM1' qui envoient des messages aux applications serveur à l'adresse 'QM2'. Les messages sur le serveur doivent être renvoyés via un autre canal à l'aide de la file d'attente de transmission 'QM1_relief' (le canal de retour par défaut serait servi avec une file d'attente de transmission 'QM1').

L'alias de file d'attente de réponse est une utilisation particulière de la définition de file d'attente éloignée nommée 'Answer_alias'. Les applications de QM1 incluent ce nom, 'Answer_alias', dans la zone de réponse de tous les messages qu'elles placent dans la file d'attente 'Inquiry'.

La définition de file d'attente de réponse 'Answer_alias' est définie comme 'Answer at QM1_relief'. Les applications sur QM1 s'attendent à ce que leurs réponses apparaissent dans la file d'attente locale nommée 'Answer'.

Les applications serveur sur QM2 utilisent la zone de réponse des messages reçus pour obtenir les noms de file d'attente et de gestionnaire de files d'attente pour les messages de réponse au demandeur sur QM1.

Définitions utilisées dans cet exemple dans QM1

L'administrateur système IBM MQ à l'adresse QM1 doit s'assurer que la file d'attente de réponse 'Answer' est créée avec les autres objets. Le nom de l'alias de gestionnaire de files d'attente, marqué par un '*', doit correspondre au nom du gestionnaire de files d'attente dans la définition de l'alias de file d'attente de réponse, également marqué par un '*'.

Objet	Définition	
File d'attente de transmission locale	QM2	
Définition de file d'attente éloignée	Nom d'objet	Requête
	Nom du gestionnaire de files d'attente éloignées	QM2
	Nom de la file d'attente distante	Requête
	Nom de la file d'attente de transmission	QM2 (VALEUR PAR DEFAUT)
Alias de gestionnaire de files d'attente	Nom d'objet	QM1_relief *
	Nom gest. de files	QM1
	Nom de la file d'attente	(blanc)
Alias de file d'attente de réponse	Nom d'objet	Alias de réponse
	Nom du gestionnaire de files d'attente éloignées	QM1_relief *
	Nom de la file d'attente distante	Réponse

Placer la définition dans QM1

Les applications remplissent les zones de réponse avec le nom d'alias de file d'attente de réponse et ne renseignent pas la zone de nom de gestionnaire de files d'attente.

Zone	Contenu
Nom de la file d'attente	Requête
Nom gest. de files	(blanc)
Nom de file d'attente de réponses	Alias de réponse
Gestionnaire de file de réponses	(blanc)

Définitions utilisées dans cet exemple à l'adresse QM2

L'administrateur système IBM MQ à l'adresse QM2 doit s'assurer que la file d'attente locale existe pour les messages entrants et que la file d'attente de transmission correctement nommée est disponible pour les messages de réponse.

Objet	Définition
File d'attente locale	Requête
File d'attente de transmission	QM1_relief

Placer la définition dans QM2

Les applications QM2 extraient le nom de la file d'attente de réponse et le nom du gestionnaire de files d'attente du message d'origine et les utilisent lors de l'insertion du message de réponse dans la file d'attente de réponse.

Zone	Contenu
Nom de la file d'attente	Réponse
Nom gest. de files	QM1_relief

Fonctionnement de l'exemple

Une explication de l'exemple et de la façon dont le gestionnaire de files d'attente utilise l'alias de file d'attente de réponse.

Dans cet exemple, les applications de demandeur sur QM1 utilisent toujours 'Answer_alias' comme file d'attente de réponse dans la zone appropriée de l'appel d'insertion. Ils extraient toujours leurs messages de la file d'attente nommée 'Answer'.

Les définitions d'alias de file d'attente de réponse peuvent être utilisées par l'administrateur système QM1 pour modifier le nom de la file d'attente de réponse 'Answer' et de la route de retour 'QM1_relief'.

La modification du nom de la file d'attente 'Answer' n'est généralement pas utile car les applications QM1 attendent leurs réponses dans cette file d'attente. Toutefois, l'administrateur système QM1 peut modifier la route de retour (classe de service), si nécessaire.

Comment le gestionnaire de files d'attente utilise l'alias de file d'attente de réponse

Le gestionnaire de files d'attente QM1 extrait les définitions de l'alias de file d'attente de réponse lorsque le nom de la file d'attente de réponse, inclus dans l'appel d'insertion par l'application, est identique à l'alias de file d'attente de réponse et que la partie du gestionnaire de files d'attente est vide.

Le gestionnaire de files d'attente remplace le nom de la file d'attente de réponse dans l'appel d'insertion par le nom de la file d'attente de la définition. Il remplace le nom de gestionnaire de files d'attente vide dans l'appel d'insertion par le nom de gestionnaire de files d'attente de la définition.

Ces noms sont portés avec le message dans le descripteur de message.

Nom de zone	Appel d'insertion	en-tête de transmission
Nom de file d'attente de réponses	Alias de réponse	Réponse
Nom du gestionnaire de files d'attente de réponse	(blanc)	QM1_relief

Présentation de l'alias de file d'attente de réponse

Une revue du processus à partir d'une application qui place un message dans une file d'attente éloignée dans la même application en supprimant le message de réponse de la file d'attente de réponse alias.

Pour compléter cet exemple, examinons le processus.

1. L'application ouvre une file d'attente nommée 'Inquiry' et y insère des messages. L'application définit les zones de réponse du descripteur de message sur:

Nom de file d'attente de réponses	Alias de réponse
Nom du gestionnaire de files d'attente de réponse	(blanc)

2. Le gestionnaire de files d'attente 'QM1' répond au nom de gestionnaire de files d'attente vide en recherchant une définition de file d'attente éloignée portant le nom 'Answer_alias'. Si aucun nom n'est trouvé, le gestionnaire de files d'attente place son propre nom, 'QM1', dans la zone du gestionnaire de files d'attente de réponse du descripteur de message.
3. Si le gestionnaire de files d'attente trouve une définition de file d'attente éloignée portant le nom 'Answer_alias', il extrait le nom de file d'attente et les noms de gestionnaire de files d'attente de la définition (nom de file d'attente = 'Answer' et nom de gestionnaire de files d'attente = 'QM1_relief'). Il les place ensuite dans les zones de réponse du descripteur de message.
4. Le gestionnaire de files d'attente 'QM1' utilise la définition de file d'attente éloignée 'Demande' pour déterminer que la file d'attente de destination prévue se trouve dans le gestionnaire de files d'attente 'QM2' et que le message est placé dans la file d'attente de transmission 'QM2'. 'QM2' est le nom de la file d'attente de transmission par défaut pour les messages destinés aux files d'attente du gestionnaire de files d'attente 'QM2'.
5. Lorsque le gestionnaire de files d'attente 'QM1' insère le message dans la file d'attente de transmission, il ajoute un en-tête de transmission au message. Cet en-tête contient le nom de la file d'attente de destination, 'Inquiry', et le gestionnaire de files d'attente de destination, 'QM2'.
6. Le message arrive dans le gestionnaire de files d'attente 'QM2' et est placé dans la file d'attente locale 'Interrogation'.
7. Une application extrait le message de cette file d'attente et le traite. L'application prépare un message de réponse et place ce message de réponse sur le nom de la file d'attente de réponse à partir du descripteur de message du message d'origine:

Nom de file d'attente de réponses	Réponse
Nom du gestionnaire de files d'attente de réponse	QM1_relief

8. Le gestionnaire de files d'attente 'QM2' exécute la commande put. Étant donné que le nom du gestionnaire de files d'attente, 'QM1_relief', est un gestionnaire de files d'attente éloignées, il place le message dans la file d'attente de transmission avec le même nom, 'QM1_relief'. Le message reçoit un en-tête de transmission contenant le nom de la file d'attente de destination, 'Answer', et le gestionnaire de files d'attente de destination, 'QM1_relief'.
9. Le message est transféré au gestionnaire de files d'attente 'QM1'. Le gestionnaire de files d'attente reconnaît que le nom de gestionnaire de files d'attente 'QM1_relief' est un alias, extrait de la définition d'alias 'QM1_relief' le nom de gestionnaire de files d'attente physique 'QM1'.
10. Le gestionnaire de files d'attente 'QM1' place ensuite le message dans le nom de file d'attente contenu dans l'en-tête de transmission, 'Answer'.
11. L'application extrait son message de réponse de la file d'attente 'Answer'.


Remarques à propos de l'utilisation en réseau

Dans un environnement de mise en file d'attente répartie, les destinations de message étant adressées uniquement avec un nom de file d'attente et un nom de gestionnaire de files d'attente, certaines règles s'appliquent.

1. Où le nom du gestionnaire de files d'attente est indiqué et le nom est différent du nom du gestionnaire de files d'attente local:

- Une file d'attente de transmission doit être disponible avec le même nom. Cette file d'attente de transmission doit faire partie d'un canal de transmission déplaçant des messages vers un autre gestionnaire de files d'attente, ou
 - Une définition d'alias de gestionnaire de files d'attente doit exister pour que le nom du gestionnaire de files d'attente puisse être résolu sur le même nom ou sur un autre nom de gestionnaire de files d'attente et une file d'attente de transmission facultative, ou
 - Si le nom de la file d'attente de transmission ne peut pas être résolu et qu'une file d'attente de transmission par défaut a été définie, la file d'attente de transmission par défaut est utilisée.
2. Lorsque seul le nom de la file d'attente est fourni, une file d'attente de tout type mais portant le même nom doit être disponible sur le gestionnaire de files d'attente local. Cette file d'attente peut être une définition de file d'attente éloignée qui se résout en: une file d'attente de transmission vers un gestionnaire de files d'attente adjacent, un nom de gestionnaire de files d'attente et une file d'attente de transmission facultative.

Pour savoir comment cela fonctionne dans un environnement de mise en cluster, voir [Clusters](#).

 Si les gestionnaires de files d'attente s'exécutent dans un groupe de partage de files d'attente (QSG) et que la mise en file d'attente intra-groupe (IGQ) est activée, vous pouvez utiliser SYSTEM.QSG.TRANSMIT.QUEUE. Pour plus d'informations, voir [Mise en file d'attente intra-groupe](#).

Prenez en compte le scénario d'un canal de transmission de messages déplaçant des messages d'un gestionnaire de files d'attente à un autre dans un environnement de mise en file d'attente répartie.

Les messages en cours de déplacement proviennent de tout autre gestionnaire de files d'attente du réseau et certains messages peuvent arriver avec un nom de gestionnaire de files d'attente inconnu comme destination. Ce problème peut se produire lorsqu'un nom de gestionnaire de files d'attente a été modifié ou supprimé du système, par exemple.

Le programme de canal reconnaît cette situation lorsqu'il ne trouve pas de file d'attente de transmission pour ces messages et place les messages dans votre file d'attente de messages non distribués (interblocage). Il est de votre responsabilité de rechercher ces messages et de faire en sorte qu'ils soient transmis à la bonne destination. Sinon, les renvoyer à l'initiateur, où l'initiateur peut être vérifié.

Des rapports d'exception sont générés dans ces circonstances, si des messages de rapport ont été demandés dans le message d'origine.

Convention de résolution de nom

La résolution de nom qui modifie l'identité de la file d'attente de destination (c'est-à-dire le changement de nom logique à physique), ne se produit qu'une seule fois et uniquement au niveau du gestionnaire de files d'attente d'origine.

L'utilisation ultérieure des différentes possibilités d'alias ne doit être utilisée que lors de la séparation et de la combinaison des flux de messages.

Routage des retours

Les messages peuvent contenir une adresse de retour sous la forme du nom d'une file d'attente et d'un gestionnaire de files d'attente. Ce formulaire d'adresse de retour peut être utilisé à la fois dans un environnement de mise en file d'attente répartie et dans un environnement de mise en cluster.

Cette adresse est normalement spécifiée par l'application qui crée le message. Il peut être modifié par n'importe quelle application qui gère ensuite le message, y compris les applications d'exit utilisateur.

Quelle que soit la source de cette adresse, toute application gérant le message peut choisir d'utiliser cette adresse pour renvoyer des messages de réponse, de statut ou de rapport à l'application d'origine.

Le mode de routage de ces messages de réponse n'est pas différent de celui du message d'origine. Vous devez savoir que les flux de messages que vous créez vers d'autres gestionnaires de files d'attente ont besoin de flux de retour correspondants.

Conflits de noms physiques

Le nom de la file d'attente de réponse de destination a été converti en nom de file d'attente physique au niveau du gestionnaire de files d'attente d'origine. Il ne doit pas être résolu à nouveau sur le gestionnaire de files d'attente qui répond.

Il est probable que des problèmes de conflit de noms ne puissent être évités que par un accord à l'échelle du réseau sur les noms de files d'attente physiques et logiques.

Gestion des conversions de noms de file d'attente

Lorsque vous créez une définition d'alias de gestionnaire de files d'attente ou une définition de file d'attente éloignée, la résolution de nom est effectuée pour chaque message portant ce nom. Cette situation doit être gérée.

Cette description est destinée aux concepteurs d'applications et aux planificateurs de canaux concernés par un système individuel comportant des canaux de message vers des systèmes adjacents. Il adopte une vue locale de la planification et du contrôle des canaux.

Lorsque vous créez une définition d'alias de gestionnaire de files d'attente ou une définition de file d'attente éloignée, la résolution de nom est effectuée pour chaque message portant ce nom, quelle que soit la source du message. Pour surveiller cette situation, qui peut impliquer un grand nombre de files d'attente dans un réseau de gestionnaires de files d'attente, vous devez conserver les tables suivantes:

- Noms des files d'attente source et des gestionnaires de files d'attente source en ce qui concerne les noms de file d'attente résolus, les noms de gestionnaire de files d'attente résolus et les noms de file d'attente de transmission résolus, avec méthode de résolution
- Noms des files d'attente source concernant:
 - Noms de file d'attente de destination résolus
 - Noms de gestionnaire de files d'attente de destination résolus
 - Files d'attente de transmission
 - Noms de canal de transmission de messages
 - Noms de système adjacents
 - Noms de file d'attente de réponse

Remarque : L'utilisation du terme *source* dans ce contexte fait référence au nom de la file d'attente ou au nom du gestionnaire de files d'attente fourni par l'application, ou à un programme de canal lors de l'ouverture d'une file d'attente pour l'insertion de messages.

Un exemple de chacune de ces tables est présenté dans [Tableau 19](#), à la page 232, [Tableau 20](#), à la page 233 et [Tableau 21](#), à la page 234.

Les noms de ces tables sont dérivés des exemples de cette section, et cette table n'est pas conçue comme un exemple pratique de résolution de nom de file d'attente dans un noeud.

File d'attente source indiquée lors de l'ouverture de la file d'attente	Gestionnaire de files d'attente source spécifié lors de l'ouverture de la file d'attente	Nom de file d'attente résolu	Nom de gestionnaire de files d'attente résolu	Nom de la file d'attente de transmission résolue	Type de résolution
QA_norm	-	QA_norm	QMB	QMB	File d'attente éloignée

Tableau 19. Résolution de nom de file d'attente sur le gestionnaire de files d'attente QMA (suite)

File d'attente source indiquée lors de l'ouverture de la file d'attente	Gestionnaire de files d'attente source spécifié lors de l'ouverture de la file d'attente	Nom de file d'attente résolu	Nom de gestionnaire de files d'attente résolu	Nom de la file d'attente de transmission résolue	Type de résolution
(n'importe quel nombre)	QMB	-	-	QMB	(aucun)
QA_norm	-	QA_norm	QMB	TX1	File d'attente éloignée
QB	QMC	QB	QMD	QMB	Alias de gestionnaire de files d'attente

Tableau 20. Résolution de nom de file d'attente au niveau du gestionnaire de files d'attente QMB

File d'attente source indiquée lors de l'ouverture de la file d'attente	Gestionnaire de files d'attente source spécifié lors de l'ouverture de la file d'attente	Nom de file d'attente résolu	Nom de gestionnaire de files d'attente résolu	Nom de la file d'attente de transmission résolue	Type de résolution
QA_norm	-	QA_norm	QMB	-	(aucun)
QA_norm	QMB	QA_norm	QMB	-	(aucun)
QA_norm	QMB_PRIORITE	QA_norm	QMB	-	Alias de gestionnaire de files d'attente
(n'importe quel nombre)	QMC	(n'importe quel nombre)	QMC	QMC	(aucun)
(n'importe quel nombre)	QMD_norme	(n'importe quel nombre)	QMD_norme	TX1	Alias de gestionnaire de files d'attente
(n'importe quel nombre)	PRIORITE QMD	(n'importe quel nombre)	PRIORITE QMD	QMD_rapide	Alias de gestionnaire de files d'attente
(n'importe quel nombre)	QMC_petite	(n'importe quel nombre)	QMC_petite	TX_petit	Alias de gestionnaire de files d'attente
(n'importe quel nombre)	QMC_large	(n'importe quel nombre)	QMC_large	TX_externe	Alias de gestionnaire de files d'attente
QB_petite	QMC	QB_petite	QMC	TX_petit	File d'attente éloignée

Tableau 20. Résolution de nom de file d'attente au niveau du gestionnaire de files d'attente QMB (suite)

File d'attente source indiquée lors de l'ouverture de la file d'attente	Gestionnaire de files d'attente source spécifié lors de l'ouverture de la file d'attente	Nom de file d'attente résolu	Nom de gestionnaire de files d'attente résolu	Nom de la file d'attente de transmission résolue	Type de résolution
QB_grande	QMC	QB_grande	QMC	TX_large	File d'attente éloignée
(n'importe quel nombre)	QME	(n'importe quel nombre)	QME	TX1	Alias de gestionnaire de files d'attente
assurance qualité	QMC	assurance qualité	QMC	TX1	File d'attente éloignée
QB	QMD	QB	QMD	TX1	File d'attente éloignée

Tableau 21. Conversion des noms de file d'attente de réponse dans le gestionnaire de files d'attente QMA

Conception d'application		Définition d'alias de réponse	
QMGR local	Nom de la file d'attente des messages	Nom d'alias de la file d'attente de réponse	Redéfini dans
QMA	QRR	QR	QRR sur QMA_class1

Numérotation des séquences de messages de canal

Le canal utilise des numéros de séquence pour vérifier que les messages sont distribués dans l'ordre dans lequel ils sont extraits de la file d'attente de transmission.

Les numéros de séquence de canal sont vérifiés lorsqu'un canal est démarré et si une non-concordance se produit, cela implique que des données de synchronisation persistantes ont été perdues de part et d'autre du canal ; par exemple, une configuration de reprise après incident (DR) ou que la fin du traitement par lots a été interrompue lorsque le canal était en attente de validation.

La réinitialisation ou l'omission des non-concordances de numéros de séquence, voir **IgnoreSeqNumberMismatch** dans la *strophe Channels* du fichier *qm.ini*, ne risque pas de perdre ou de dupliquer un lot de messages et ne réinitialise pas l'état en attente de validation d'un canal.

Ces informations peuvent être affichées à l'aide de la commande `DISPLAY CHSTATUS`. Le numéro de séquence et un identificateur appelé LUWID sont stockés en mémoire persistante pour le dernier message transféré dans un lot. Ces valeurs sont utilisées lors du démarrage du canal pour s'assurer que les deux extrémités de la liaison sont d'accord sur les messages qui ont été transférés avec succès.

Extraction séquentielle des messages

Si une application insère une séquence de messages dans la même file d'attente de destination, ces messages peuvent être extraits en séquence par une application **unique** avec une séquence d'opérations MQGET, si les conditions suivantes sont remplies:

- Toutes les demandes d'insertion ont été effectuées à partir de la même application.
- Toutes les demandes d'insertion provenaient de la même unité de travail ou toutes les demandes d'insertion étaient effectuées en dehors d'une unité de travail.
- Les messages ont tous la même priorité.

- Les messages ont tous la même persistance.
- Pour la mise en file d'attente distante, la configuration est telle qu'il ne peut y avoir qu'un seul chemin depuis l'application qui effectue la demande d'insertion, via son gestionnaire de files d'attente, via l'intercommunication, vers le gestionnaire de files d'attente de destination et la file d'attente cible.
- Les messages ne sont pas insérés dans une file d'attente de rebut (par exemple, si une file d'attente est temporairement saturée).
- L'application qui reçoit le message ne modifie pas délibérément l'ordre d'extraction, par exemple en spécifiant un *MsgId* ou un *CorrelId* particulier ou en utilisant des priorités de message.
- Une seule application effectue des opérations d'extraction pour extraire les messages de la file d'attente de destination. S'il existe plusieurs applications, ces applications doivent être conçues pour obtenir tous les messages de chaque séquence insérés par une application d'envoi.

Remarque : Les messages provenant d'autres tâches et unités de travail peuvent être entrecoupés avec la séquence, même si la séquence a été placée à partir d'une seule unité de travail.

Si ces conditions ne peuvent pas être remplies et que l'ordre des messages dans la file d'attente cible est important, l'application peut être codée pour utiliser son propre numéro de séquence de message dans le cadre du message afin d'assurer l'ordre des messages.

Séquence d'extraction de messages rapides et non persistants

Les messages non persistants sur un canal rapide peuvent dépasser les messages persistants sur le même canal et donc arriver hors séquence. L'agent MCA récepteur place immédiatement les messages non persistants dans la file d'attente de destination et les rend visibles. Les messages persistants ne sont pas rendus visibles jusqu'au point de synchronisation suivant.

Test de bouclage

Le *test de bouclage* est une technique sur Multiplatforms qui permet de tester une liaison de communication sans établir de liaison avec une autre machine.

Vous configurez une connexion entre deux gestionnaires de files d'attente comme s'ils se trouvent sur des machines distinctes, mais vous testez la connexion en vous reconnectant à un autre processus sur la même machine. Cette technique signifie que vous pouvez tester votre code de communication sans avoir besoin d'un réseau actif.

La façon dont vous le faites dépend des produits et des protocoles que vous utilisez.

Sur les systèmes Windows , vous pouvez utiliser l'adaptateur "loopback".

Pour plus d'informations, reportez-vous à la documentation des produits que vous utilisez.

Traçage de route et enregistrement d'activité

Vous pouvez confirmer la route qu'un message emprunte à travers une série de gestionnaires de files d'attente de deux manières.

Vous pouvez utiliser l'application d'affichage de route IBM MQ , disponible via la commande de contrôle **dspmqrte**, ou vous pouvez utiliser l'enregistrement d'activité. Ces deux rubriques sont décrites dans [Référence de surveillance](#).

Introduction à la gestion de files d'attente réparties

La gestion de files d'attente réparties (DQM) permet de définir et de contrôler la communication entre les gestionnaires de files d'attente.

Gestion des files d'attente réparties:

- Permet de définir et de contrôler les canaux de communication entre les gestionnaires de files d'attente
- Fournit un service de canal de transmission permettant de déplacer des messages d'un type de *file d'attente locale*, appelée file d'attente de transmission, vers des liaisons de communication sur un

système local et des liaisons de communication vers des files d'attente locales sur un gestionnaire de files d'attente de destination.




- Fournit des fonctions permettant de surveiller le fonctionnement des canaux et de diagnostiquer les problèmes, à l'aide de panneaux, de commandes et de programmes

Les définitions de canal associent des noms de canal à des files d'attente de transmission, des identificateurs de liaison de communication et des attributs de canal. Les définitions de canal sont implémentées de différentes manières sur différentes plateformes. L'envoi et la réception de messages sont contrôlés par des programmes appelés *agents MCA* (Message Channel Agent), qui utilisent les définitions de canal pour démarrer et contrôler la communication.





Les MCM sont à leur tour contrôlés par le DQM lui-même. La structure dépend de la plateforme, mais inclut généralement des programmes d'écoute et des moniteurs de déclenchement, ainsi que des commandes et des panneaux de l'opérateur.

Un *canal de transmission de messages* est un canal unidirectionnel permettant de déplacer des messages d'un gestionnaire de files d'attente vers un autre. Ainsi, un canal de message comporte deux points d'extrémité, représentés par une paire de MCM. Chaque noeud final possède une définition de son extrémité du canal de transmission de messages. Par exemple, une extrémité définit un expéditeur, l'autre un destinataire.

Pour plus de détails sur la définition des canaux, voir:

-  [«Surveillance et contrôle des canaux sur AIX, Linux, and Windows»](#), à la page 268
-  [«Monitoring and controlling channels on z/OS»](#), à la page 1013
-  [«Surveillance et contrôle des canaux sur IBM i»](#), à la page 293

Pour obtenir des exemples de planification de canal de message, voir:

-  [Exemple de planification de canal de message pour AIX, Linux, and Windows](#)
-  [Exemple de planification de canal de message pour IBM i](#)
-  [Exemple de planification de canal de message pour z/OS](#)
-  [Exemple de planification de canal de message pour z/OS à l'aide de groupes de partage de files d'attente](#)

Pour plus d'informations sur les exits de canal, voir [Programmes d'exit de canal pour les canaux de messagerie](#).

Concepts associés

[«Envoi et réception de messages»](#), à la page 237

La figure suivante illustre le modèle de gestion de files d'attente réparties, détaillant les relations entre les entités lors de la transmission de messages. Il montre également le flux pour le contrôle.

[«Fonction de contrôle de canal»](#), à la page 244

La fonction de contrôle de canal fournit des fonctions permettant de définir, de surveiller et de contrôler des canaux.

[«Que se passe-t-il lorsqu'un message ne peut pas être distribué?»](#), à la page 259

Lorsqu'un message ne peut pas être distribué, l'agent MCA peut le traiter de plusieurs manières. Il peut réessayer, renvoyer à l'expéditeur ou le placer dans la file d'attente des messages non livrés.

[«Fichiers d'initialisation et de configuration»](#), à la page 264

Le traitement des données d'initialisation de canal dépend de votre plateforme IBM MQ .

[«Conversion de données pour les messages»](#), à la page 265

Les messages IBM MQ peuvent nécessiter une conversion de données lorsqu'ils sont envoyés entre des files d'attente de différents gestionnaires de files d'attente.

[«Ecriture de vos propres agents MCA»](#), à la page 266

IBM MQ vous permet d'écrire vos propres programmes MCA (Message Channel Agent) ou d'en installer un à partir d'un fournisseur de logiciels indépendant.

«Autres éléments à prendre en compte pour la gestion de files d'attente réparties», à la page 266
 Autres rubriques à prendre en compte lors de la préparation de IBM MQ pour la gestion de files d'attente réparties. Cette rubrique traite des files d'attente de messages non distribuées, des files d'attente en cours d'utilisation, des extensions système et des programmes d'exit utilisateur, ainsi que de l'exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées.

Référence associée

Exemple d'informations de configuration

Envoi et réception de messages

La figure suivante illustre le modèle de gestion de files d'attente réparties, détaillant les relations entre les entités lors de la transmission de messages. Il montre également le flux pour le contrôle.

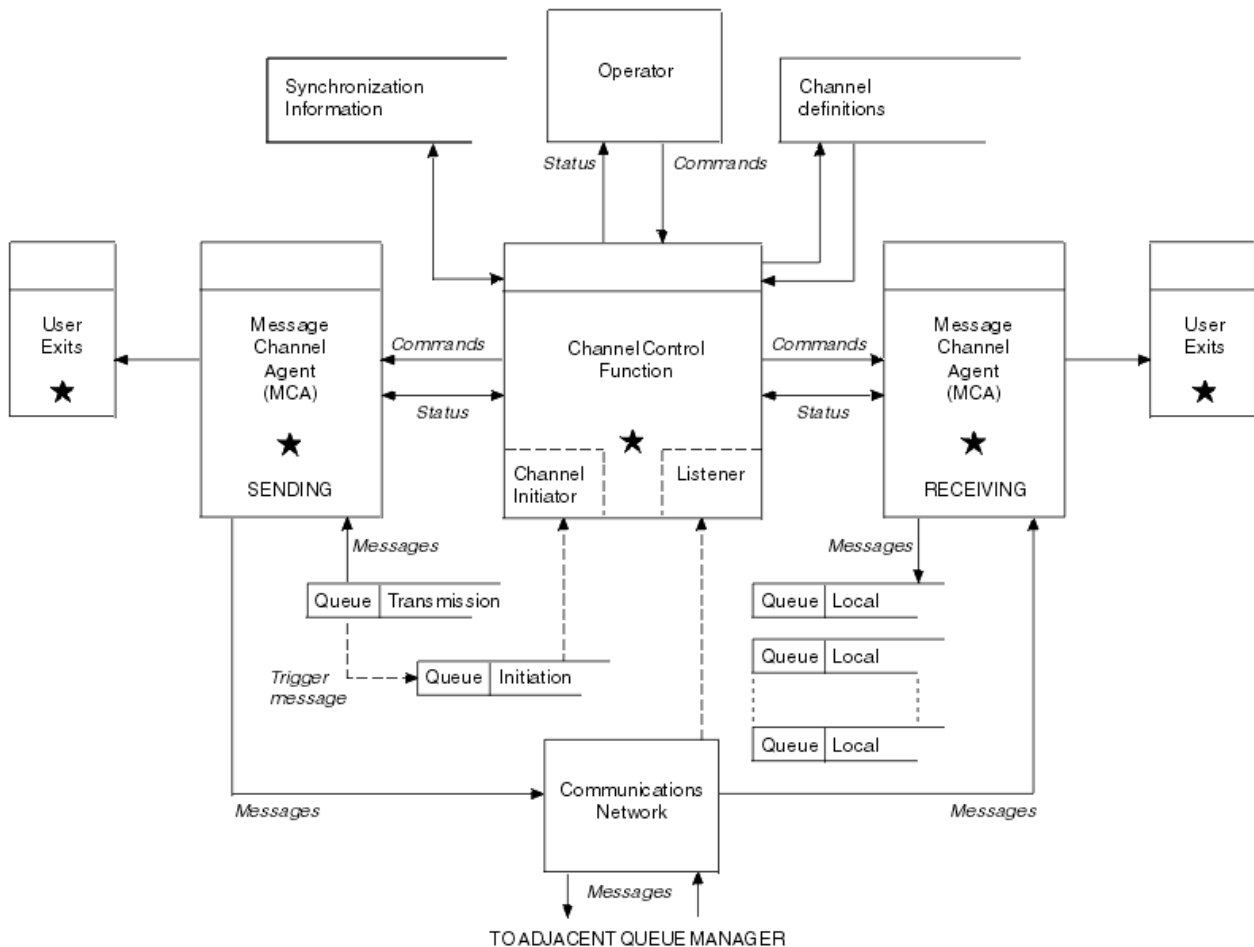


Figure 15. Modèle de gestion de files d'attente réparties

Remarque :

1. Il existe un agent MCA par canal, en fonction de la plateforme. Il peut exister une ou plusieurs fonctions de contrôle de canal pour un gestionnaire de files d'attente particulier.
2. L'implémentation des MCA et des fonctions de contrôle de canal dépend fortement de la plateforme. Il peut s'agir de programmes ou de processus ou d'unités d'exécution, et il peut s'agir d'une seule entité ou de plusieurs parties indépendantes ou liées.
3. Tous les composants marqués d'une étoile peuvent utiliser l'interface MQI.

Paramètres du canal

Un agent MCA reçoit ses paramètres de l'une des manières suivantes:

- S'il est démarré par une commande, le nom de canal est transmis dans une zone de données. L'agent MCA lit ensuite directement la définition de canal pour obtenir ses attributs.
- Pour les canaux émetteurs et, dans certains cas, serveurs, l'agent MCA peut être démarré automatiquement par le déclencheur du gestionnaire de files d'attente. Le nom de canal est extrait de la définition de processus de déclenchement, le cas échéant, et est transmis à l'agent MCA. Le traitement restant est le même que celui décrit précédemment. Les canaux de serveur ne doivent être configurés pour être déclenchés que s'ils sont qualifiés complets, c'est-à-dire qu'ils spécifient un nom de connexion (CONNNAME) auquel se connecter.
- S'il est démarré à distance par un émetteur, un serveur, un demandeur ou une connexion client, le nom de canal est transmis dans les données initiales de l'agent MCA partenaire. L'agent MCA lit directement la définition de canal pour obtenir ses attributs.

Certains attributs non définis dans la définition de canal sont également négociables:

Fractionner les messages

Si une extrémité ne prend pas en charge les messages fractionnés, les messages fractionnés ne sont pas envoyés.

Capacité de conversion

Si une extrémité ne peut pas effectuer la conversion de page de codes nécessaire ou la conversion de codage numérique si nécessaire, l'autre extrémité doit la traiter. Si aucune des deux extrémités ne la prend en charge, si nécessaire, le canal ne peut pas démarrer.

Prise en charge de la liste de diffusion

Si une extrémité ne prend pas en charge les listes de distribution, l'agent MCA partenaire définit un indicateur dans sa file d'attente de transmission pour qu'il sache intercepter les messages destinés à plusieurs destinations.

Statut du canal et numéros de séquence

Les programmes de l'agent MCA conservent les enregistrements du numéro de séquence en cours et du numéro d'unité d'oeuvre logique pour chaque canal, ainsi que l'état général du canal. Certaines plateformes vous permettent d'afficher ces informations d'état pour vous aider à contrôler les canaux.

Comment envoyer un message à un autre gestionnaire de files d'attente

Cette section décrit le moyen le plus simple d'envoyer un message entre les gestionnaires de files d'attente, y compris les prérequis et les autorisations requises. D'autres méthodes peuvent également être utilisées pour envoyer des messages à un gestionnaire de files d'attente éloignées.


Avant d'envoyer un message d'un gestionnaire de files d'attente à un autre, vous devez effectuer les étapes suivantes:

1. Vérifiez que le protocole de communication choisi est disponible.
2. Démarrer les gestionnaires de files d'attente.
3. Démarrez les initiateurs de canal.
4. Démarrez les programmes d'écoute.



Vous devez également disposer de l'autorisation de sécurité IBM MQ appropriée pour créer les objets requis.

Pour envoyer des messages d'un gestionnaire de files d'attente à un autre:

- Définissez les objets suivants sur le gestionnaire de files d'attente source:
 - Canal émetteur
 - Définition de file d'attente éloignée

- File d'attente d'initialisation ( obligatoire sous z/OS, sinon facultatif)
- File d'attente de transmission
- File d'attente de rebut
- Définissez les objets suivants sur le gestionnaire de files d'attente cible:
 - Canal récepteur
 - File d'attente cible
 - File d'attente de rebut

Vous pouvez utiliser plusieurs méthodes pour définir ces objets, en fonction de votre plateforme IBM MQ :

- Sur toutes les plateformes, vous pouvez utiliser les commandes de script IBM MQ (MQSC) décrites dans [Les commandes MQSC](#) les commandes PCF (Programmable Command Format) décrites dans [Automating administration tasks](#) ou dans IBM MQ Explorer.
-  Sous z/OS, vous pouvez également utiliser les panneaux d'opération et de contrôle décrits dans [Administration d' IBM MQ for z/OS](#) .
-  Sous IBM i, vous pouvez également utiliser l'interface de panneau.

Pour plus d'informations sur la création des composants permettant d'envoyer des messages à un autre gestionnaire de files d'attente, voir les sous-rubriques suivantes:

Concepts associés

[«Techniques de mise en file d'attente répartie IBM MQ», à la page 213](#)

Les sous-rubriques de cette section décrivent les techniques utilisées lors de la planification des canaux. Ces sous-rubriques décrivent les techniques permettant de planifier la connexion de vos gestionnaires de files d'attente et de gérer le flux de messages entre vos applications.

[«Introduction à la gestion de files d'attente réparties», à la page 235](#)

La gestion de files d'attente réparties (DQM) permet de définir et de contrôler la communication entre les gestionnaires de files d'attente.

[«Canaux de déclenchement», à la page 260](#)

IBM MQ fournit une fonction permettant de démarrer automatiquement une application lorsque certaines conditions d'une file d'attente sont remplies. Cette fonction est appelée déclenchement.

[«Sécurité des messages», à la page 258](#)

Outre les fonctions de récupération standard d' IBM MQ, la gestion des files d'attente réparties garantit que les messages sont distribués correctement à l'aide d'une procédure de point de synchronisation coordonnée entre les deux extrémités du canal de transmission de messages. Si cette procédure détecte une erreur, elle ferme le canal afin que vous puissiez examiner le problème et conserve les messages en toute sécurité dans la file d'attente de transmission jusqu'à ce que le canal soit redémarré.

Tâches associées

[«Création de gestionnaires de files d'attente sur Multiplatforms», à la page 7](#)

Avant de pouvoir utiliser des messages et des files d'attente, vous devez créer et démarrer au moins un gestionnaire de files d'attente et ses objets associés. Un gestionnaire de files d'attente gère les ressources qui lui sont associées, en particulier les files d'attente qu'il possède. Il fournit des services de mise en file d'attente aux applications pour les appels et les commandes MQI (Message queuing Interface) permettant de créer, de modifier, d'afficher et de supprimer des objets IBM MQ .

[«Surveillance et contrôle des canaux sur AIX, Linux, and Windows», à la page 268](#)

Pour DQM, vous devez créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Vous pouvez contrôler les canaux à l'aide de commandes, de programmes, de IBM MQ Explorer, de fichiers pour les définitions de canal et d'une zone de stockage pour les informations de synchronisation.

[«Surveillance et contrôle des canaux sur IBM i», à la page 293](#)

Utilisez les commandes et les panneaux DQM pour créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Chaque gestionnaire de files d'attente dispose d'un

programme DQM permettant de contrôler les interconnexions aux gestionnaires de files d'attente éloignées compatibles.

«[Configuration des connexions entre le client et le serveur](#)», à la page 17

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

«[Configuration d'un cluster de gestionnaires de files d'attente](#)», à la page 316

Les clusters fournissent un mécanisme d'interconnexion des gestionnaires de files d'attente qui simplifie à la fois la configuration initiale et la gestion en cours. Vous pouvez définir des composants de cluster et créer et gérer des clusters.

«[Setting up communications with other queue managers on z/OS](#)», à la page 1009

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Définition des canaux

Pour envoyer des messages d'un gestionnaire de files d'attente à un autre, vous devez définir deux canaux. Vous devez définir un canal sur le gestionnaire de files d'attente source et un canal sur le gestionnaire de files d'attente cible.

Sur le gestionnaire de files d'attente source

Définissez un canal avec un type de canal SENDER. Vous devez spécifier les éléments suivants:

- Nom de la file d'attente de transmission à utiliser (attribut XMITQ).
- Nom de connexion du système partenaire (attribut CONNAME).
- Nom du protocole de communication que vous utilisez (attribut TRPTYPE). Sous IBM MQ for z/OS, le protocole doit être TCP ou LU6.2. Sur Multiplatforms, il n'est pas nécessaire de spécifier le protocole. Vous pouvez le laisser pour extraire la valeur de votre définition de canal par défaut.

Les détails de tous les attributs de canal sont indiqués dans [Attributs de canal](#).

Sur le gestionnaire de files d'attente cible

Définissez un canal avec un type de canal RECEIVER et le même nom que le canal émetteur.

Indiquez le nom du protocole de communication que vous utilisez (attribut TRPTYPE). Sous IBM MQ for z/OS, le protocole doit être TCP ou LU6.2. Sur Multiplatforms, il n'est pas nécessaire de spécifier le protocole. Vous pouvez le laisser pour extraire la valeur de votre définition de canal par défaut.

Les définitions de canal récepteur peuvent être génériques. Cela signifie que si plusieurs gestionnaires de files d'attente communiquent avec le même récepteur, les canaux émetteurs peuvent tous spécifier le même nom pour le récepteur et une définition de récepteur s'applique à tous.

Une fois que vous avez défini le canal, vous pouvez le tester à l'aide de la commande PING CHANNEL. Cette commande envoie un message spécial du canal émetteur au canal récepteur et vérifie qu'il est renvoyé.

Remarque : La valeur du paramètre TRPTYPE est ignorée par l'agent MCA qui répond. Par exemple, un TRPTYPE de TCP sur la définition de canal émetteur démarre avec un TRPTYPE de LU62 sur la définition de canal récepteur en tant que partenaire.

Définition des files d'attente

Pour envoyer des messages d'un gestionnaire de files d'attente à un autre, vous devez définir jusqu'à six files d'attente. Vous devez définir jusqu'à quatre files d'attente sur le gestionnaire de files d'attente source et jusqu'à deux files d'attente sur le gestionnaire de files d'attente cible.

Sur le gestionnaire de files d'attente source

- Définition de file d'attente éloignée

Dans cette définition, spécifiez les éléments suivants:

Nom du gestionnaire de files d'attente éloignées

Nom du gestionnaire de files d'attente cible.


Nom de la file d'attente distante


Nom de la file d'attente cible sur le gestionnaire de files d'attente cible.

Nom de la file d'attente de transmission


Nom de la file d'attente de transmission. Il n'est pas nécessaire d'indiquer ce nom de file d'attente de transmission. Si vous ne le faites pas, une file d'attente de transmission portant le même nom que le gestionnaire de files d'attente cible est utilisée. S'il n'existe pas, la file d'attente de transmission par défaut est utilisée. Il est conseillé d'attribuer à la file d'attente de transmission le même nom que le gestionnaire de files d'attente cible afin que la file d'attente soit trouvée par défaut.

- Définition de file d'attente d'initialisation

 Opération requise. Vous devez utiliser la file d'attente d'initialisation appelée SYSTEM.CHANNEL.INITQ.

 Facultatif. Pensez à nommer la file d'attente d'initialisation SYSTEM.CHANNEL.INITQ.

- Définition de file d'attente de transmission

Une file d'attente locale avec l'attribut USAGE défini sur XMITQ.  Si vous utilisez l'interface native IBM MQ for IBM i, l'attribut USAGE est *TMQ.

- Définition de file d'attente de rebut

Définissez une file d'attente de rebut dans laquelle les messages non distribués peuvent être écrits.

Sur le gestionnaire de files d'attente cible

- Définition de file d'attente locale

File d'attente cible. Le nom de cette file d'attente doit être identique à celui indiqué dans la zone de nom de file d'attente éloignée de la définition de file d'attente éloignée sur le gestionnaire de files d'attente source.

- Définition de file d'attente de rebut

Définissez une file d'attente de rebut dans laquelle les messages non distribués peuvent être écrits.

Concepts associés

[«Création d'une file d'attente de transmission», à la page 241](#)

Pour qu'un canal (autre qu'un canal demandeur) puisse être démarré, la file d'attente de transmission doit être définie comme décrit dans cette section. La file d'attente de transmission doit être nommée dans la définition de canal.

[«Création d'une file d'attente de transmission sous IBM i», à la page 242](#)

Vous pouvez créer une file d'attente de transmission sur la plateforme IBM i à l'aide du panneau Create MQM Queue.

Création d'une file d'attente de transmission

Pour qu'un canal (autre qu'un canal demandeur) puisse être démarré, la file d'attente de transmission doit être définie comme décrit dans cette section. La file d'attente de transmission doit être nommée dans la définition de canal.

Définissez une file d'attente locale avec l'attribut USAGE défini sur XMITQ pour chaque canal de transmission de messages. Si vous souhaitez utiliser une file d'attente de transmission spécifique dans vos définitions de file d'attente éloignée, créez une file d'attente éloignée comme indiqué.

Pour créer une file d'attente de transmission, utilisez les commandes IBM MQ (MQSC), comme illustré dans les exemples suivants:

Exemple de création de file d'attente de transmission

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') USAGE(XMITQ)
```

Exemple de création de file d'attente éloignée

```
DEFINE QREMOTE(PAYROLL) DESCR('Remote queue for QM2') +  
XMITQ(QM2) RNAME(PAYROLL) RQMNAME(QM2)
```

Pensez à nommer la file d'attente de transmission en tant que nom de gestionnaire de files d'attente sur le système distant, comme illustré dans les exemples.

Création d'une file d'attente de transmission sous IBM i

Vous pouvez créer une file d'attente de transmission sur la plateforme IBM i à l'aide du panneau Create MQM Queue.

Vous devez définir une file d'attente locale avec l'attribut de zone Utilisation défini sur *TMQ, pour chaque canal de transmission de messages.

Si vous souhaitez utiliser des définitions de file d'attente éloignée, utilisez la même commande pour créer une file d'attente de type *RMT et d'utilisation de *NORMAL.

Pour créer une file d'attente de transmission, utilisez la commande CRTMQMQ à partir de la ligne de commande pour afficher le premier panneau de création de file d'attente ; voir [Figure 16, à la page 242](#).

```
Create MQM Queue (CRTMQMQ)  
Type choices, press Enter.  
Queue name . . . . .  
Queue type . . . . . ____ *ALS, *LCL, *MDL, *RMT  
Message Queue Manager name . . . *DFT_____
```

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

Figure 16. Créer une file d'attente (1)

Entrez le nom de la file d'attente et indiquez le type de file d'attente à créer: local, distant ou alias. Pour une file d'attente de transmission, indiquez Local (*LCL) sur ce panneau et appuyez sur Entrée.

La deuxième page du panneau Création d'une file d'attente MQM s'affiche ; voir [Figure 17, à la page 243](#).

```

Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Queue name . . . . . > HURS.2.HURS.PRIORIT

Queue type . . . . . > *LCL      *ALS, *LCL, *MDL, *RMT
Message Queue Manager name . . . *DFT
Replace . . . . . *NO      *NO, *YES
Text 'description' . . . . . '
Put enabled . . . . . *YES    *SYSDFTQ, *NO, *YES
Default message priority . . . . 0      0-9, *SYSDFTQ
Default message persistence . . *NO    *SYSDFTQ, *NO, *YES
Process name . . . . . '
Triggering enabled . . . . . *NO    *SYSDFTQ, *NO, *YES
Get enabled . . . . . *YES    *SYSDFTQ, *NO, *YES
Sharing enabled . . . . . *YES    *SYSDFTQ, *NO, *YES

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

```

Figure 17. Créer une file d'attente (2)

Modifiez l'une des valeurs par défaut affichées. Appuyez sur la page vers le bas pour accéder à l'écran suivant ; voir [Figure 18](#), à la page 243.

```

Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Default share option . . . . . *YES    *SYSDFTQ, *NO, *YES
Message delivery sequence . . . *PTY    *SYSDFTQ, *PTY, *FIFO
Harden backout count . . . . . *NO    *SYSDFTQ, *NO, *YES
Trigger type . . . . . *FIRST  *SYSDFTQ, *FIRST, *ALL...
Trigger depth . . . . . 1      1-999999999, *SYSDFTQ
Trigger message priority . . . . 0      0-9, *SYSDFTQ
Trigger data . . . . . '
Retention interval . . . . . 999999999 0-999999999, *SYSDFTQ
Maximum queue depth . . . . . 5000   1-24000, *SYSDFTQ
Maximum message length . . . . . 4194304 0-4194304, *SYSDFTQ
Backout threshold . . . . . 0      0-999999999, *SYSDFTQ
Backout requeue queue . . . . . '
Initiation queue . . . . . '

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

```

Figure 18. Créer une file d'attente (3)

Entrez *TMQ pour la file d'attente de transmission dans la zone Utilisation de ce panneau et modifiez les valeurs par défaut indiquées dans les autres zones.

```

Create MQM Queue (CRTMQMQ)

Type choices, press Enter.

Usage . . . . . *TMQ      *SYSDFTQ, *NORMAL, *TMQ
Queue depth high threshold . . . 80      0-100, *SYSDFTQ
Queue depth low threshold . . . 20      0-100, *SYSDFTQ
Queue full events enabled . . . *YES   *SYSDFTQ, *NO, *YES
Queue high events enabled . . . *YES   *SYSDFTQ, *NO, *YES
Queue low events enabled . . . *YES   *SYSDFTQ, *NO, *YES
Service interval . . . . . 999999999 0-999999999, *SYSDFTQ
Service interval events . . . *NONE  *SYSDFTQ, *HIGH, *OK, *NONE
Distribution list support . . . *NO    *SYSDFTQ, *NO, *YES
Cluster Name . . . . . *SYSDFTQ
Cluster Name List . . . . . *SYSDFTQ
Default Binding . . . . . *SYSDFTQ *SYSDFTQ, *OPEN, *NOTFIXED

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

```

Figure 19. Créer une file d'attente (4)

Lorsque vous vous êtes assuré que les zones contiennent les données correctes, appuyez sur Entrée pour créer la file d'attente.

Démarrage du canal

Lorsque vous placez des messages dans la file d'attente éloignée définie sur le gestionnaire de files d'attente source, ils sont stockés dans la file d'attente de transmission jusqu'à ce que le canal soit démarré. Une fois le canal démarré, les messages sont distribués à la file d'attente cible sur le gestionnaire de files d'attente éloignées.

Démarrez le canal sur le gestionnaire de files d'attente émetteur à l'aide de la commande START CHANNEL. Lorsque vous démarrez le canal émetteur, le canal récepteur est démarré automatiquement (par le programme d'écoute) et les messages sont envoyés à la file d'attente cible. Les deux extrémités du canal de transmission de messages doivent être en cours d'exécution pour que les messages puissent être transférés.

Etant donné que les deux extrémités du canal se trouvent sur des gestionnaires de files d'attente différents, elles peuvent avoir été définies avec des attributs différents. Pour résoudre les différences, il existe une négociation de données initiale entre les deux extrémités lorsque le canal démarre. En général, les deux extrémités du canal fonctionnent avec les attributs qui ont besoin de moins de ressources. Cela permet aux systèmes de plus grande taille de prendre en charge les ressources moindres des systèmes de plus petite taille à l'autre extrémité du canal de transmission de messages.

L'agent MCA émetteur fractionne les messages volumineux avant de les envoyer via le canal. Ils sont réassemblés au niveau du gestionnaire de files d'attente éloignées. Cela n'est pas évident pour l'utilisateur.

Un agent MCA peut transférer des messages à l'aide de plusieurs unités d'exécution. Ce processus, appelé *pipeline*, permet à l'agent MCA de transférer les messages plus efficacement, avec moins d'états d'attente. Le pipelining améliore les performances des canaux.

Fonction de contrôle de canal

La fonction de contrôle de canal fournit des fonctions permettant de définir, de surveiller et de contrôler des canaux.

Les commandes sont émises via des panneaux, des programmes ou à partir d'une ligne de commande vers la fonction de contrôle de canal. L'interface du panneau affiche également le statut du canal et les données de définition de canal. Vous pouvez utiliser les formats de commande programmables ou les

commandes IBM MQ (MQSC) et les commandes de contrôle détaillées dans [«Surveillance et contrôle des canaux sur AIX, Linux, and Windows»](#), à la page 268.

Les commandes appartiennent aux groupes suivants:

- Administration des canaux
- Contrôle de canal
- Surveillance du statut du canal

Les commandes d'administration de canal traitent des définitions des canaux. Ils vous permettent d'effectuer les opérations suivantes:

- Créer une définition de canal
- Copie d'une définition de canal
- Modification d'une définition de canal
- Suppression d'une définition de canal

Les commandes de contrôle de canal gèrent le fonctionnement des canaux. Ils vous permettent d'effectuer les opérations suivantes:

- Démarrage d'un canal
- Arrêter un canal
- Resynchroniser avec le partenaire (dans certaines implémentations)
- Réinitialiser les numéros de séquence de message
- Résoudre un lot de messages en attente de validation
- Ping ; envoi d'une communication test sur le canal

La surveillance des canaux affiche l'état des canaux, par exemple:

- Paramètres de canal en cours
- Indique si le canal est actif ou inactif
- Indique si le canal s'est arrêté dans un état synchronisé

Concepts associés

[Où trouver des informations pour vous aider à identifier les problèmes](#)

Préparation des canaux

Avant de tenter de démarrer un canal de transmission de messages ou un canal MQI, vous devez préparer le canal. Vous devez vous assurer que tous les attributs des définitions de canal local et distant sont corrects et compatibles.

Les [attributs de canal](#) décrivent les définitions et les attributs de canal.

Bien que vous ayez configuré des définitions de canal explicites, les négociations de canal effectuées au démarrage d'un canal peuvent remplacer l'une ou l'autre des valeurs définies. Ce comportement est normal, et n'est pas apparent pour l'utilisateur, et a été organisé de cette manière de sorte que des définitions autrement incompatibles puissent fonctionner ensemble.

Définition automatique des canaux de réception et de connexion serveur

Dans IBM MQ on Multiplatforms, s'il n'existe pas de définition de canal appropriée, pour un canal récepteur ou de connexion serveur pour lequel la définition automatique est activée, une définition est créée automatiquement. La définition est créée à l'aide de:

1. La définition de canal de modèle appropriée, SYSTEM.AUTO.RECEIVER ou SYSTEM.AUTO.SVRCONN. Les définitions de canal de modèle pour la définition automatique sont identiques aux valeurs par défaut du système, SYSTEM.DEF.RECEIVER et SYSTEM.DEF.SVRCONN, à l'exception de la zone de description, qui est "défini automatiquement par", suivie de 49 blancs. L'administrateur système peut choisir de modifier n'importe quelle partie des définitions de canal de modèle fournies.
2. Informations provenant du système partenaire. Les valeurs du partenaire sont utilisées pour le nom de canal et la valeur de retour à la ligne du numéro de séquence.
3. Un programme d'exit de canal, que vous pouvez utiliser pour modifier les valeurs créées par la définition automatique. Voir [Programme d'exit de définition automatique de canal](#).

La description est ensuite vérifiée pour déterminer si elle a été modifiée par un exit de définition automatique ou si la définition de modèle a été modifiée. Si les 44 premiers caractères sont toujours "Auto-defined by" suivi de 29 blancs, le nom du gestionnaire de files d'attente est ajouté. Si les 20 derniers caractères sont toujours vides, l'heure locale et la date sont ajoutées.

Lorsque la définition a été créée et stockée, le démarrage du canal se poursuit comme si la définition avait toujours existé. La taille du lot, la taille de la transmission et la taille du message sont négociées avec le partenaire.

Définition d'autres objets

Pour qu'un canal de transmission de messages puisse être démarré, les deux extrémités doivent être définies (ou activées pour la définition automatique) sur leurs gestionnaires de files d'attente. La file d'attente de transmission qu'elle doit servir doit être définie pour le gestionnaire de files d'attente à l'extrémité émettrice. La liaison de communication doit être définie et disponible. Il peut être nécessaire de préparer d'autres objets IBM MQ, tels que des définitions de file d'attente éloignée, des définitions d'alias de gestionnaire de files d'attente et des définitions d'alias de file d'attente de réponse, pour implémenter les scénarios décrits dans [«Configuration de la file d'attente répartie»](#), à la page 212.

Pour plus d'informations sur la définition de canaux MQI, voir [«Définition de canaux MQI»](#), à la page 32.

Plusieurs canaux de transmission par file d'attente de transmission

Il est possible de définir plus d'un canal par file d'attente de transmission, mais un seul de ces canaux peut être actif à la fois. Envisagez cette option pour la mise à disposition de routes alternatives entre les gestionnaires de files d'attente pour l'équilibrage du trafic et l'action corrective en cas d'échec de liaison. Une file d'attente de transmission ne peut pas être utilisée par un autre canal si le canal précédent à utiliser s'est arrêté en laissant un lot de messages en attente de validation à l'extrémité émettrice. Pour plus d'informations, voir [«Traitement des canaux en attente de validation»](#), à la page 257.

Démarrage d'un canal

Un canal peut être amené à commencer à transmettre des messages de l'une des quatre manières suivantes. Il peut être :

- Démarré par un opérateur (non récepteur, récepteur de cluster ou canaux de connexion serveur).
- Déclenché à partir de la file d'attente de transmission. Cette méthode s'applique aux canaux émetteurs et aux canaux serveur qualifiés complets (ceux qui spécifient un CONNAME) uniquement. Vous devez préparer les objets nécessaires pour le déclenchement des canaux.
- Lancé à partir d'un programme d'application (non récepteur, récepteur de cluster ou canaux de connexion serveur).
- Démarré à distance à partir du réseau par un émetteur, un émetteur de cluster, un demandeur, un serveur ou un canal de connexion client. Le récepteur, le récepteur de cluster et éventuellement les transmissions de canal serveur et demandeur sont démarrés de cette manière, de même que les canaux de connexion serveur. Les canaux eux-mêmes doivent déjà être démarrés (c'est-à-dire activés).

Remarque : Etant donné qu'un canal est démarré, il ne s'agit pas nécessairement de transmettre des messages. Au lieu de cela, il peut être activé pour démarrer la transmission lorsque l'un des quatre événements décrits précédemment se produit. L'activation et la désactivation d'un canal sont réalisées à l'aide des commandes de l'opérateur START et STOP.

Etats des canaux

Un canal peut être dans l'un des nombreux états à tout moment. Certains états ont également des sous-états. A partir d'un état donné, un canal peut passer à d'autres états.

Figure 20, à la page 247 affiche la hiérarchie de tous les états de canal possibles et les sous-états qui s'appliquent à chacun des états de canal.

La Figure 21, à la page 248 présente les liens entre les états de canal. Ces liens s'appliquent à tous les types de canal de transmission de messages et de canaux de connexion serveur.

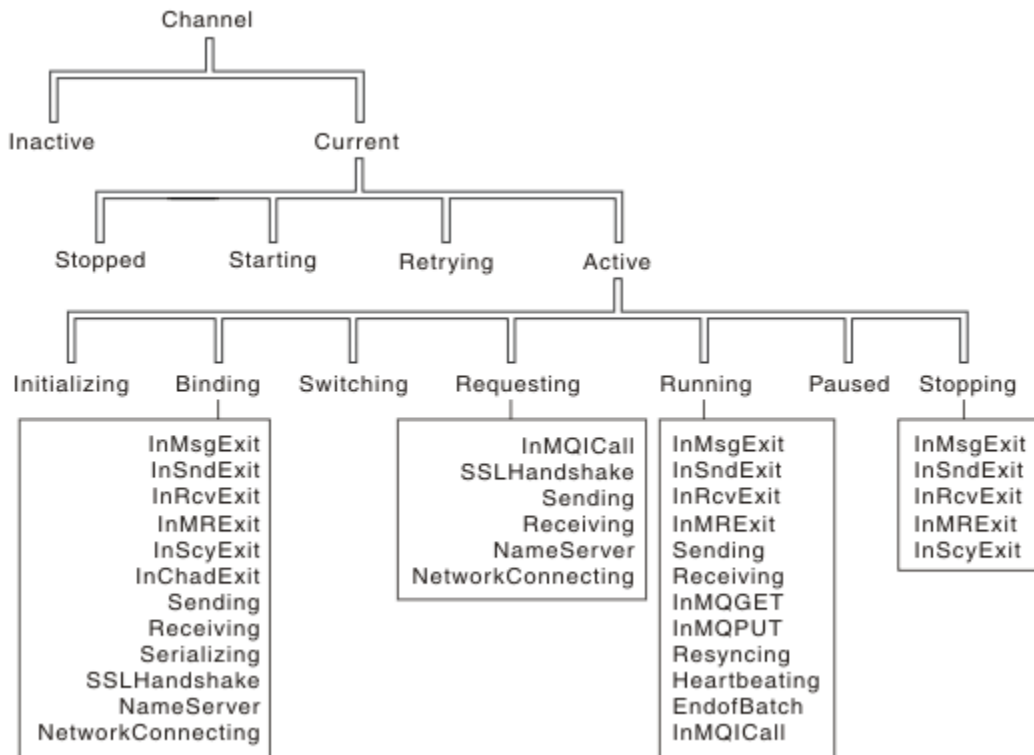


Figure 20. Etats et sous-états des canaux

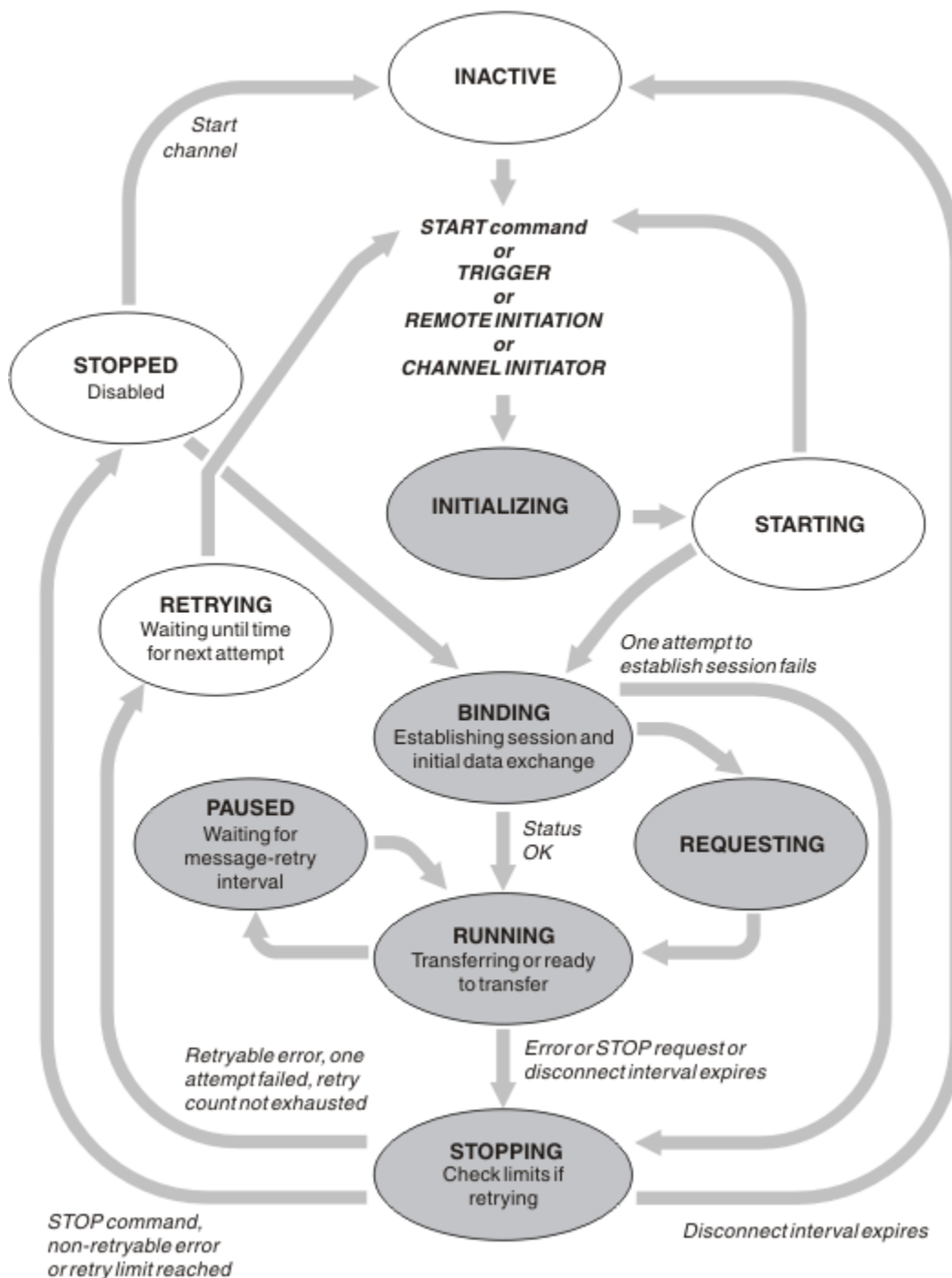


Figure 21. Flux entre les états de canal

En cours et actif

Un canal est *en cours* s'il est dans un état autre que inactif. Un canal en cours est *actif* sauf s'il est à l'état REESSAYEZ, STOPPED ou STARTING. Lorsqu'un canal est actif, il consomme des ressources et un processus ou une unité d'exécution est en cours d'exécution. Les sept états possibles d'un canal actif (INITIALISATION, LIAISON, COMMUTATION, DEMANDE, EXÉCUTION, PAUSED ou ARRET) sont mis en évidence dans Figure 21, à la page 248.

Un canal actif peut également afficher un sous-état fournissant plus de détails sur ce que le canal fait exactement. Les sous-états de chaque état sont indiqués dans le Figure 20, à la page 247.

En cours et actif

Le canal est "en cours" s'il est dans un état autre que inactif. Un canal en cours est "actif" sauf s'il est à l'état REESSAYEZ, STOPPED ou STARTING.

Si un canal est "actif", il peut également afficher un sous-état donnant plus de détails sur ce que fait exactement le canal.

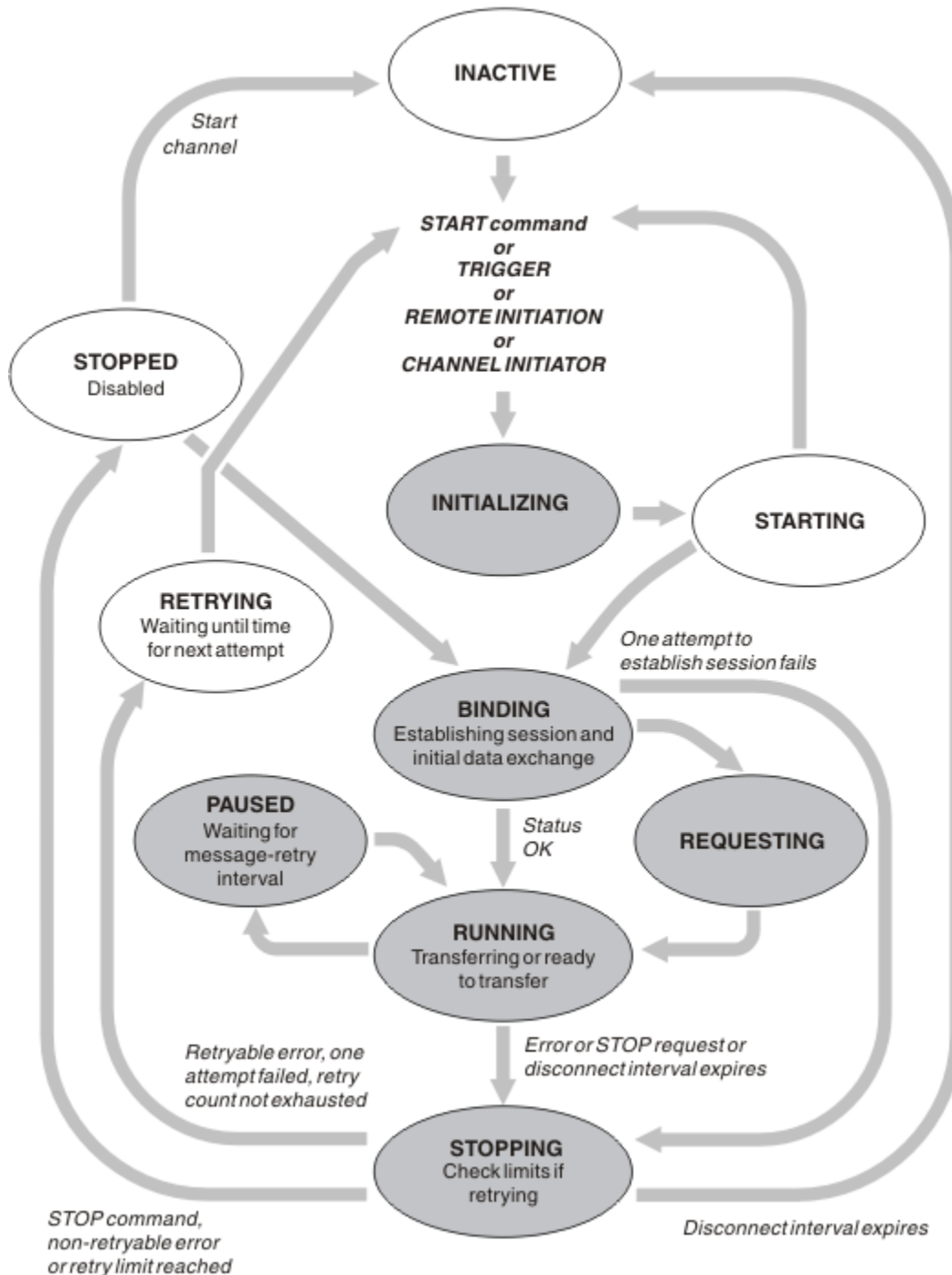


Figure 22. Flux entre les états de canal

Remarque :

1. Lorsqu'un canal se trouve dans l'un des six états mis en évidence dans Figure 22, à la page 249 (INITIALIZING, BINDING, REQUÉRIR, RUNNING, PAUSED ou STOPPING), il consomme des ressources et un processus ou une unité d'exécution est en cours d'exécution ; le canal est *actif*.

2. Lorsqu'un canal est à l'état STOPPED, la session peut être active car l'état suivant n'est pas encore connu.

Spécification du nombre maximal de canaux en cours

Vous pouvez spécifier le nombre maximal de canaux pouvant être en cours simultanément. Ce nombre correspond au nombre de canaux ayant des entrées dans la table de statut des canaux, y compris les canaux en cours de relance et les canaux arrêtés. Spécifiez ceci pour votre plateforme:

- ▶ **z/OS** Utilisez la commande ALTER QMGR MAXCHL .
- ▶ **IBM i** Editez le fichier d'initialisation du gestionnaire de files d'attente.
- ▶ **Linux** ▶ **AIX** Editez le fichier de configuration du gestionnaire de files d'attente.
- Utilisez la console IBM MQ Explorer.

Pour plus d'informations sur les valeurs définies à l'aide du fichier d'initialisation ou de configuration, voir [Strophes du fichier de configuration pour la mise en file d'attente répartie](#). Pour plus d'informations sur la spécification du nombre maximal de canaux, voir les rubriques suivantes:

- ▶ **ALW** [Administration d' IBM MQ](#).
- ▶ **IBM i** [Administration d' IBM MQ for IBM i](#).
- ▶ **z/OS** [Administration d' IBM MQ for z/OS](#).

Remarque :

1. Les canaux de connexion serveur sont inclus dans ce nombre.
2. Un canal doit être à jour pour pouvoir devenir actif. Si un canal est démarré, mais ne peut pas devenir actif, le démarrage échoue.

Spécification du nombre maximal de canaux actifs

Vous pouvez également spécifier le nombre maximal de canaux actifs pour éviter que votre système ne soit surchargé par de nombreux canaux de démarrage. Si vous utilisez cette méthode, définissez l'attribut d'intervalle de déconnexion sur une valeur faible pour permettre aux canaux en attente de démarrer dès que d'autres canaux s'arrêtent.

Chaque fois qu'un canal tente à nouveau d'établir une connexion avec son partenaire, il doit devenir un canal actif. Si la tentative échoue, le canal en cours reste inactif jusqu'à la prochaine tentative. Nombre de nouvelles tentatives d'un canal et fréquence, déterminé par le nombre de nouvelles tentatives et les attributs d'intervalle entre les nouvelles tentatives. Il existe des valeurs courtes et longues pour ces deux attributs. Pour plus d'informations, voir [Attributs de canal](#) .

Lorsqu'un canal doit devenir un canal actif (parce qu'une commande START a été émise, ou parce qu'elle a été déclenchée, ou parce qu'il est l'heure d'une nouvelle tentative), mais qu'il ne peut pas le faire car le nombre de canaux actifs est déjà à la valeur maximale, le canal attend que l'un des emplacements actifs soit libéré par une autre instance de canal qui cesse d'être active. Toutefois, si un canal est en cours de démarrage parce qu'il est en cours de lancement à distance et qu'aucun emplacement actif n'est disponible pour ce canal à ce moment-là, le lancement à distance est rejeté.

Chaque fois qu'un canal, autre qu'un canal demandeur, tente de devenir actif, il passe à l'état STARTING. Cet état se produit même si un créneau actif est immédiatement disponible, bien qu'il ne soit à l'état STARTING que pendant un court laps de temps. Cependant, si le canal doit attendre un emplacement actif, il est à l'état STARTING pendant qu'il est en attente.

Les canaux demandeurs ne passent pas à l'état STARTING. Si un canal demandeur ne peut pas démarrer car le nombre de canaux actifs est déjà à la limite, le canal se termine de manière anormale.

Chaque fois qu'un canal, autre qu'un canal demandeur, ne parvient pas à obtenir un emplacement actif et en attend donc un, un message est consigné dans le journal ▶ **z/OS** ou dans la console z/OS , et

un événement est généré. Lorsqu'un attribut est libéré ultérieurement et que le canal est en mesure de l'acquiescer, un autre message et un autre événement sont générés. Aucun de ces événements et messages n'est généré si le canal est en mesure d'acquiescer un créneau immédiatement.

Si une commande STOP CHANNEL est émise alors que le canal est en attente de devenir actif, il passe à l'état STOPPED. Un événement Channel-Stopped est émis.

Les canaux de connexion serveur sont inclus dans le nombre maximal de canaux actifs.

Pour plus d'informations sur la spécification du nombre maximal de canaux actifs, voir les rubriques suivantes:

- ▶ **ALW** Administration d' IBM MQ.
- ▶ **IBM i** Administration d' IBM MQ for IBM i.
- ▶ **z/OS** Administration d' IBM MQ for z/OS.

Erreurs de canal

Les erreurs sur les canaux entraînent l'arrêt des autres transmissions par le canal. Si le canal est un émetteur ou un serveur, il passe à l'état RETRY car il est possible que le problème se soit résolu lui-même. S'il ne peut pas passer à l'état RETRY, le canal passe à l'état STOPPED.

Pour les canaux d'envoi, la file d'attente de transmission associée est définie sur GET (DISABLED) et le déclenchement est désactivé. (Une commande STOP avec STATUS (STOPPED) prend le côté qui l'a émise à l'état STOPPED ; seule l'expiration de l'intervalle de déconnexion ou une commande STOP avec STATUS (INACTIVE) l'arrête normalement et devient inactive.) Les canaux qui sont à l'état STOPPED ont besoin d'une intervention de l'opérateur avant de pouvoir redémarrer (voir «Redémarrage des canaux arrêtés», à la page 256).

Remarque : Pour les systèmes **IBM i** IBM i, AIX, Linux, and Windows , un initiateur de canal doit être en cours d'exécution pour qu'une nouvelle tentative soit effectuée. Si l'initiateur de canal n'est pas disponible, le canal devient inactif et doit être redémarré manuellement. Si vous utilisez un script pour démarrer le canal, assurez-vous que l'initiateur de canal est en cours d'exécution avant d'essayer d'exécuter le script.

Long retry count (LONGRTY) décrit le fonctionnement de la relance. Si l'erreur disparaît, le canal redémarre automatiquement et la file d'attente de transmission est réactivée. Si le nombre maximal de nouvelles tentatives est atteint sans l'effacement de l'erreur, le canal passe à l'état ARRETE. Un canal arrêté doit être redémarré manuellement par l'opérateur. Si l'erreur est toujours présente, elle ne fait pas de nouvelle tentative. Lorsqu'elle démarre correctement, la file d'attente de transmission est réactivée.

▶ **z/OS** Si l'initiateur de canal s'arrête alors qu'un canal est à l'état REESSAYEZ ou STOPPED, l'état du canal est mémorisé lors du redémarrage de l'initiateur de canal. Toutefois, le statut du canal SVRCONN est réinitialisé si l'initiateur de canal s'arrête alors que le canal est à l'état STOPPED.

▶ **Multi** Si le gestionnaire de files d'attente s'arrête alors qu'un canal est à l'état REESSAYEZ ou STOPPED, l'état du canal est mémorisé lors du redémarrage du gestionnaire de files d'attente. A partir de la IBM MQ 8.0 , cela s'applique également aux canaux SVRCONN. Auparavant, le statut de canal pour le type de canal SVRCONN était réinitialisé si l'initiateur de canal était arrêté alors que le canal était à l'état STOPPED.

Si un canal ne parvient pas à insérer un message dans la file d'attente cible car cette file d'attente est saturée ou bloquée, il peut relancer l'opération un certain nombre de fois (spécifié dans l'attribut de nombre de relances de message) à un intervalle de temps (spécifié dans l'attribut d'intervalle de relances de message). Vous pouvez également écrire votre propre exit de relance de message qui détermine les circonstances à l'origine d'une nouvelle tentative et le nombre de tentatives effectuées. Le canal passe à l'état PAUSED en attendant la fin de l'intervalle entre les relances de message.

Voir Attributs de canal pour plus d'informations sur les attributs de canal et Programmes d'exit de canal pour les canaux de messagerie pour plus d'informations sur l'exit de relance de message.

Nombre maximal de canaux de connexion serveur

Vous pouvez définir des limites de canal de connexion serveur pour empêcher les applications client d'épuiser les ressources de canal du gestionnaire de files d'attente avec le paramètre **MAXINST** et pour empêcher une application client unique d'épuiser la capacité de canal de connexion serveur avec le paramètre **MAXINSTC**.

Vous définissez **MAXINST** et **MAXINSTC** à l'aide de la commande **DEFINE CHANNEL**.

Un nombre total maximal de canaux peut être actif à tout moment sur un gestionnaire de files d'attente individuel. Le nombre total d'instances de canal de connexion serveur est inclus dans le nombre maximal de canaux actifs.

Si vous ne spécifiez pas le nombre maximal d'instances simultanées d'un canal de connexion serveur pouvant être démarrées, il est possible qu'une application client unique, se connectant à un canal de connexion serveur unique, épuise le nombre maximal de canaux actifs disponibles. Lorsque le nombre maximal de canaux actifs est atteint, il empêche le démarrage d'autres canaux sur le gestionnaire de files d'attente. Pour éviter cette situation, vous devez limiter le nombre d'instances simultanées d'un canal de connexion serveur individuel pouvant être démarrées, quel que soit le client qui les a démarrées.

Si la valeur de la limite est inférieure au nombre d'instances en cours d'exécution du canal de connexion serveur, même à zéro, les canaux en cours d'exécution ne sont pas affectés. Les nouvelles instances ne peuvent pas être démarrées tant qu'un nombre suffisant d'instances existantes n'ont pas cessé de s'exécuter, de sorte que le nombre d'instances en cours d'exécution est inférieur à la valeur de la limite.

En outre, de nombreux canaux de connexion client différents peuvent se connecter à un canal de connexion serveur individuel. La limite du nombre d'instances simultanées d'un canal de connexion serveur individuel pouvant être démarrées, quel que soit le client qui les a démarrées, empêche tout client d'épuiser la capacité maximale du canal actif du gestionnaire de files d'attente. Si vous ne limitez pas également le nombre d'instances simultanées d'un canal de connexion serveur individuel pouvant être démarrées à partir d'un client individuel, il est possible qu'une seule application client défaillante ouvre un nombre de connexions tel qu'elle épuise la capacité de canal allouée à un canal de connexion serveur individuel et empêche donc les autres clients qui ont besoin d'utiliser le canal de se connecter à ce canal. Pour éviter cette situation, vous devez limiter le nombre d'instances simultanées d'un canal de connexion serveur individuel pouvant être démarrées à partir d'un client individuel.

Si la valeur de la limite de client individuel est réduite en dessous du nombre d'instances du canal de connexion serveur qui s'exécutent actuellement à partir de clients individuels, même à zéro, les canaux en cours d'exécution ne sont pas affectés. Toutefois, les nouvelles instances du canal de connexion serveur ne peuvent pas être démarrées à partir d'un client individuel qui dépasse la nouvelle limite jusqu'à ce qu'un nombre suffisant d'instances existantes de ce client aient cessé de s'exécuter de sorte que le nombre d'instances en cours d'exécution soit inférieur à la valeur de ce paramètre.

Référence associée

[Attributs de canal et types de canal](#)

[De la définition d'un canal](#)

Vérification de la disponibilité de l'autre extrémité du canal

Vous pouvez utiliser l'intervalle des pulsations, l'intervalle des signaux de présence et le délai de réception pour vérifier que l'autre extrémité du canal est disponible.

Pulsations

Vous pouvez utiliser l'attribut de canal d'intervalle des pulsations pour indiquer que les flux doivent être transmis à partir de l'agent MCA émetteur lorsqu'il n'y a pas de message dans la file d'attente de transmission, comme décrit dans [Intervalle des pulsations \(HBINT\)](#).

Signal de présence

 Sous z/OS, si vous utilisez TCP/IP comme protocole de transport, vous pouvez également indiquer une valeur pour l'attribut de canal d'intervalle **Keepalive (KAINT)**. Il est recommandé

d'attribuer à l'intervalle **Keepalive** une valeur supérieure à l'intervalle des pulsations et une valeur inférieure à la valeur de déconnexion. Vous pouvez utiliser cet attribut pour spécifier une valeur de délai d'attente pour chaque canal, comme décrit dans [Intervalle de signal de présence \(KAIN\)](#).

Multi Sur les systèmes IBM i, AIX, Linux, and Windows, si vous utilisez TCP comme protocole de transport, vous pouvez définir `keepalive=yes`. Si vous spécifiez cette option, TCP vérifie périodiquement que l'autre extrémité de la connexion est toujours disponible. Ce n'est pas le cas, le canal est arrêté. Cette option est décrite dans [Intervalle de signal de présence \(KAIN\)](#).

Si vous avez des canaux non fiables qui signalent des erreurs TCP, l'utilisation de l'option **Keepalive** signifie que vos canaux sont plus susceptibles de récupérer.

Vous pouvez spécifier des intervalles de temps pour contrôler le comportement de l'option **Keepalive**. Lorsque vous modifiez l'intervalle de temps, seuls les canaux TCP/IP démarrés après la modification sont affectés. Assurez-vous que la valeur que vous choisissez pour l'intervalle de temps est inférieure à la valeur de l'intervalle de déconnexion pour le canal.

Pour plus d'informations sur l'utilisation de l'option **Keepalive**, voir le paramètre [KAIN](#) dans la commande **DEFINE CHANNEL**.

Délai de réception

Si vous utilisez TCP comme protocole de transport, l'extrémité de réception d'une connexion de canal non MQI inactive est également fermée si aucune donnée n'est reçue pendant un certain temps. Cette période, la valeur *délai de réception*, est déterminée en fonction de la valeur **HBINT** (intervalle des pulsations).

Sur les systèmes IBM MQ for IBM i, AIX, Linux, and Windows, la valeur *receive time-out* est définie comme suit:

1. Pour un nombre initial de flux, avant toute négociation, la valeur *délai d'attente de réception* est le double de la valeur **HBINT** de la définition de canal.
2. Une fois que les canaux ont négocié une valeur **HBINT**, si **HBINT** est défini sur une valeur inférieure à 60 secondes, la valeur *délai d'attente de réception* est définie sur deux fois cette valeur. Si **HBINT** est défini sur 60 secondes ou plus, la valeur *délai d'attente de réception* est supérieure de 60 secondes à la valeur de **HBINT**.

z/OS Sous z/OS, la valeur *receive time-out* est définie comme suit:

1. Pour un nombre initial de flux, avant toute négociation, la valeur *délai d'attente de réception* est le double de la valeur **HBINT** de la définition de canal.
2. Si **RCVTIME** est défini, le délai d'attente est défini sur l'une des valeurs suivantes, en fonction du paramètre **RCVTYPE**, et soumis à toute limite imposée par **RCVTIME** s'il s'applique:
 - **HBINT** négocié multiplié par une constante
 - **HBINT** négocié plus un nombre constant de secondes
 - Nombre constant de secondes

RCVTIME ne s'applique pas lorsque **RCVTYPE (EQUAL)** est configuré. Si vous utilisez la valeur constante **RCVTIME** et un intervalle des pulsations, n'indiquez pas de valeur **RCVTIME** inférieure à l'intervalle des pulsations. Pour plus de détails sur les attributs **RCVTIME**, **RCVTIME** et **RCVTYPE**, voir la commande [ALTER QMGR](#).

Remarque :

1. Si l'une des valeurs est zéro, il n'y a pas de délai d'attente.
2. Pour les connexions qui ne prennent pas en charge les pulsations, la valeur **HBINT** est négociée à zéro à l'étape 2 et, par conséquent, il n'y a pas de délai d'attente. Vous devez donc utiliser TCP/IP **KEEPALIVE**.

3. Pour les connexions client qui utilisent le partage de conversations, les pulsations peuvent transiter à travers le canal (à partir des deux extrémités) tout le temps, et pas seulement lorsqu'une requête MQGET est en attente.
4. Pour les connexions client pour lesquelles le partage de conversations n'est pas utilisé, les signaux de présence sont transmis à partir du serveur uniquement lorsque le client émet un appel MQGET avec attente. Par conséquent, il n'est pas recommandé de définir un intervalle de pulsation trop petit pour les canaux client. Par exemple, si le signal de présence est défini sur 10 secondes, un appel MQCMIT échoue (avec MQRC_CONNECTION_BROKEN) s'il prend plus de 20 secondes à valider car aucune donnée n'a été collectée pendant cette période. Cela peut se produire avec de grandes unités de travail. Toutefois, cela ne se produit pas si des valeurs appropriées sont choisies pour l'intervalle des pulsations car seule la commande MQGET avec attente prend beaucoup de temps.
Si **SHARECNV** n'est pas égal à zéro, le client utilise une connexion en duplex intégral, ce qui signifie que le client peut (et effectue) un signal de présence lors de tous les appels MQI
5. L'annulation de la connexion après deux fois l'intervalle des pulsations est valide car un flux de données ou de pulsations est attendu au moins à chaque intervalle des pulsations. Toutefois, si l'intervalle des pulsations est trop petit, des problèmes peuvent se produire, en particulier si vous utilisez des exits de canal. Par exemple, si la valeur **HBINT** est d'une seconde et qu'un exit d'émission ou de réception est utilisé, la fin de réception attend seulement 2 secondes avant d'annuler le canal. Si l'agent MCA exécute une tâche telle que le chiffrement du message, cette valeur peut être trop courte.

Paramètres suggérés

IBM MQ for z/OS

Comme point de départ initial, vous pouvez utiliser:

```
/cpl ALTER QMGR TCPKEEP(YES) RCVTTYPE(ADD) RCVTIME(60) ADOPTMCA(ALL) ADOPTCHK(ALL)
```

où cpl est le préfixe de commande du sous-système de gestionnaire de files d'attente.

Pour plus d'informations sur les différents paramètres, voir [ALTER QMGR](#) et [Disponibilité du réseau IBM MQ](#).

Si l'adresse IP de l'expéditeur peut être traduite en plusieurs adresses, vous devrez peut-être définir **ADOPTCHK** sur QMNAME au lieu de ALL.

IBM MQ for Multiplatforms

Dans `qm.ini`, ajoutez les informations suivantes:

```
TCP:
KeepAlive=Yes
CHANNELS:
AdoptNewMCA=ALL
AdoptNewMCACheck=ALL
```

Pour plus d'informations, voir [ALTER QMGR](#), strophes du fichier de configuration pour la mise en file d'attente répartie et «Strophe channels du fichier `qm.ini`», à la page 131.

Si l'adresse IP de l'expéditeur peut être traduite en plusieurs adresses, vous devrez peut-être définir **AdoptNewMCACheck** sur QMNAME au lieu de **ALL**.

Adoption d'un agent MCA

La fonction d'adoption d'agent MCA permet à IBM MQ d'annuler un canal récepteur et d'en démarrer un nouveau à sa place.

Si un canal perd le contact, le canal récepteur peut être laissé dans un état de "réception des communications". Lorsque les communications sont rétablies, le canal émetteur tente de se reconnecter. Si le gestionnaire de files d'attente éloignées détecte que le canal récepteur est déjà en cours d'exécution,


il n'autorise pas le démarrage d'une autre version du même canal récepteur. Ce problème nécessite une intervention de l'utilisateur pour corriger le problème ou l'utilisation du signal de présence du système.

La fonction d'adoption d'un agent MCA résout automatiquement le problème. Il permet à IBM MQ d'annuler un canal récepteur et d'en démarrer un nouveau à sa place.

Tâches associées

Administration de IBM MQ

 Administration de IBM MQ for z/OS

 Administration de IBM MQ for IBM i



Arrêt et mise au repos des canaux

Vous pouvez arrêter et mettre au repos un canal avant l'expiration de l'intervalle de temps de déconnexion.


Les canaux de transmission de messages sont conçus pour être des connexions à exécution longue entre les gestionnaires de files d'attente avec un arrêt ordonné contrôlé uniquement par l'attribut de canal d'intervalle de déconnexion. Ce mécanisme fonctionne bien sauf si l'opérateur doit arrêter le canal avant l'expiration de l'intervalle de temps de déconnexion. Ce besoin peut se produire dans les situations suivantes:

- Mise au repos du système
- Conservation des ressources
- Action unilatérale à une extrémité d'un canal

Dans ce cas, vous pouvez arrêter le canal. Pour ce faire, procédez comme suit:

- la commande STOP CHANNEL MQSC
- la commande Stop Channel PCF
- l'explorateur IBM MQ
-   autres mécanismes spécifiques à la plateforme, comme suit:

 **Pour z/OS :**
Le panneau Stop a channel

 **Pour IBM i :**
La commande CL ENDMQMCHL ou l'option END sur le panneau WRKMQMCHL

Il existe trois options pour arrêter les canaux à l'aide des commandes suivantes:

QUIESCE

L'option QUIESCE tente d'arrêter le lot de messages en cours avant d'arrêter le canal.


FORCE

L'option FORCE tente d'arrêter le canal immédiatement et peut nécessiter une resynchronisation du canal lors de son redémarrage car le canal peut être laissé en attente de validation.

 Sous IBM MQ for z/OS, FORCE interrompt toute réallocation de message en cours, ce qui peut laisser les messages BIND_NOT_FIXED partiellement réalloués ou dans l'ordre.

TERMINATE

L'option TERMINATE tente d'arrêter le canal immédiatement et arrête l'unité d'exécution ou le processus du canal.

 Sous IBM MQ for z/OS, TERMINATE interrompt toute réallocation de message en cours, ce qui peut laisser les messages BIND_NOT_FIXED partiellement réalloués ou dans le désordre.

Toutes ces options laissent le canal à l'état STOPPED, ce qui nécessite l'intervention de l'opérateur pour le redémarrer.

L'arrêt du canal à l'extrémité émettrice est effectif mais nécessite une intervention de l'opérateur pour redémarrer. A l'extrémité réceptrice du canal, les choses sont beaucoup plus difficiles car l'agent MCA attend les données du côté émetteur et il n'est pas possible de lancer un arrêt *ordonné* du canal du côté récepteur ; la commande d'arrêt est en attente jusqu'à ce que l'agent MCA revienne de son attente de données.

Par conséquent, il existe trois façons recommandées d'utiliser les canaux, en fonction des caractéristiques opérationnelles requises:

- Si vous souhaitez que vos canaux soient à exécution longue, notez qu'il ne peut y avoir de fin ordonnée qu'à partir de la fin de l'envoi. Lorsque des canaux sont interrompus, c'est-à-dire arrêtés, une intervention de l'opérateur (commande START CHANNEL) est requise pour les redémarrer.
- Si vous souhaitez que vos canaux soient actifs uniquement lorsqu'il y a des messages à transmettre, définissez l'intervalle de déconnexion sur une valeur assez faible. Le paramètre par défaut est élevé et n'est donc pas recommandé pour les canaux où ce niveau de contrôle est requis. Etant donné qu'il est difficile d'interrompre le canal récepteur, l'option la plus économique consiste à déconnecter et reconnecter automatiquement le canal en fonction de la charge de travail. Pour la plupart des canaux, la définition appropriée de l'intervalle de déconnexion peut être établie de manière heuristique.
- Vous pouvez utiliser l'attribut heartbeat-interval pour que l'agent MCA émetteur envoie un flux de pulsations à l'agent MCA récepteur pendant les périodes pendant lesquelles il n'a pas de messages à envoyer. Cette action libère l'agent MCA récepteur de son état d'attente et lui permet de mettre le canal au repos sans attendre l'expiration de l'intervalle de déconnexion. Attribuez à l'intervalle des pulsations une valeur inférieure à celle de l'intervalle de déconnexion.

Remarque :

1. Il est conseillé de définir un intervalle de déconnexion faible ou d'utiliser des signaux de présence pour les canaux serveur. Cette valeur faible permet de tenir compte du cas où le canal demandeur se termine de manière anormale (par exemple, parce que le canal a été annulé) lorsqu'il n'y a pas de messages à envoyer par le canal serveur. Si l'intervalle de déconnexion est défini comme étant élevé et que les signaux de présence ne sont pas utilisés, le serveur ne détecte pas que le demandeur s'est arrêté (ce qu'il ne fera que la prochaine fois qu'il essaiera d'envoyer un message au demandeur). Alors que le serveur est toujours en cours d'exécution, il maintient la file d'attente de transmission ouverte pour une entrée exclusive afin d'obtenir d'autres messages qui arrivent dans la file d'attente. Si une tentative de redémarrage du canal est effectuée à partir du demandeur, la demande de démarrage reçoit une erreur car la file d'attente de transmission est toujours ouverte pour une entrée exclusive sur le serveur. Il est nécessaire d'arrêter le canal serveur, puis de le redémarrer à partir du demandeur.

Redémarrage des canaux arrêtés

Lorsqu'un canal passe à l'état STOPPED, vous devez le redémarrer manuellement.



Pourquoi et quand exécuter cette tâche

Pour les canaux émetteurs ou serveurs, lorsque le canal est passé à l'état STOPPED, la file d'attente de transmission associée a été définie sur GET (DISABLED) et le déclenchement a été désactivé. Lorsque la demande de démarrage est reçue, ces attributs sont réinitialisés automatiquement.

z/OS Si l'initiateur de canal s'arrête alors qu'un canal est à l'état REESSAYEZ ou STOPPED, l'état du canal est mémorisé lors du redémarrage de l'initiateur de canal. Toutefois, le statut du canal SVRCONN est réinitialisé si l'initiateur de canal s'arrête alors que le canal est à l'état STOPPED.

Multi Si le gestionnaire de files d'attente s'arrête alors qu'un canal est à l'état REESSAYEZ ou STOPPED, l'état du canal est mémorisé lors du redémarrage du gestionnaire de files d'attente. A partir de la IBM MQ 8.0, cela s'applique également aux canaux SVRCONN. Auparavant, le statut de canal pour le type de canal SVRCONN était réinitialisé si l'initiateur de canal était arrêté alors que le canal était à l'état STOPPED.

Procédure

- Redémarrez le canal de l'une des manières suivantes:
 - A l'aide de la [commande START CHANNEL MQSC](#).
 - A l'aide de la [commande PCF Démarrer un canal](#).
 - A l'aide de [IBM MQ Explorer](#)
 -  Sous z/OS, à l'aide du [panneau Démarrer un canal](#).
 -  Sous IBM i, à l'aide de la [commande CL STRMQMCHL](#) ou de l'option START du [panneau WRKMQMCHL](#).

Traitement des canaux en attente de validation

Un canal en attente de validation est un canal en attente de validation avec un canal distant sur lequel des messages ont été envoyés et reçus.

Pourquoi et quand exécuter cette tâche

Notez la distinction entre ce gestionnaire de files d'attente et un gestionnaire de files d'attente en attente de savoir quels messages doivent être validés dans une file d'attente.

Vous pouvez réduire la possibilité pour un canal d'être mis en attente de validation à l'aide du paramètre de canal de pulsation par lots (**BATCHHB**). Lorsqu'une valeur est spécifiée pour ce paramètre, un canal émetteur vérifie que le canal distant est toujours actif avant de prendre toute autre mesure. Si aucune réponse n'est reçue, le canal récepteur est considéré comme n'étant plus actif. Les messages peuvent être annulés et réacheminés, et le canal émetteur n'est pas mis en attente de validation. Cela réduit le temps pendant lequel le canal peut être mis en attente de la période entre le moment où le canal émetteur vérifie que le canal récepteur est toujours actif et celui où le canal récepteur a reçu les messages émis. Pour plus d'informations sur le paramètre de signal de présence par lots, voir [Attributs de canal](#).

Les problèmes de canal en attente de validation sont généralement résolus automatiquement. Même lorsque la communication est perdue et qu'un canal est mis en attente de validation avec un lot de messages à l'expéditeur dont le statut de réception est inconnu, la situation est résolue lorsque la communication est rétablie. Les numéros de séquence et les enregistrements LUWID sont conservés à cette fin. Le canal est en attente de validation jusqu'à ce que les informations LUWID aient été échangées, et un seul lot de messages peut être en attente de validation pour le canal.

Vous pouvez, si nécessaire, resynchroniser le canal manuellement. Le terme manuel inclut l'utilisation d'opérateurs ou de programmes qui contiennent des commandes de gestion du système IBM MQ. Le processus de resynchronisation manuelle fonctionne comme suit. Cette description utilise des commandes MQSC, mais vous pouvez également utiliser les équivalents PCF.

Procédure

1. La commande **DISPLAY CHSTATUS** permet de rechercher l'ID de l'unité d'oeuvre logique (LUWID) validée en dernier pour chaque côté du canal.

Pour ce faire, utilisez les commandes suivantes:

- Pour le côté en attente de validation du canal:

```
DISPLAY CHSTATUS(name) SAVED CURLUWID
```

Vous pouvez utiliser les paramètres **CONNAME** et **XMITQ** pour identifier plus précisément le canal.

- Pour la partie réceptrice du canal:

```
DISPLAY CHSTATUS( name ) SAVED LSTLUWID
```

Vous pouvez utiliser le paramètre **CONNAME** pour identifier plus précisément le canal.

Remarque : Les commandes sont différentes car seul le côté émetteur du canal peut être en attente de validation. Le côté récepteur n'est jamais en doute.

IBM i Sous IBM i, la commande **DISPLAY CHSTATUS** peut être exécutée à partir d'un fichier à l'aide de la commande **STRMQMMQSC** ou de la commande CL Work with MQM Channel Status, **WRKMQMCHST**.

2. Si les deux LUWID sont identiques, utilisez la commande **RESOLVE CHANNEL** pour valider les messages en attente de validation.

Si les deux LUWID sont identiques, le destinataire a validé l'unité de travail que l'expéditeur considère comme en attente de validation. Le côté émetteur peut désormais supprimer les messages en attente de validation de la file d'attente de transmission et les réactiver. Cette opération est effectuée à l'aide de la commande **RESOLVE CHANNEL** suivante:

```
RESOLVE CHANNEL(name) ACTION(COMMIT)
```

3. Si les deux LUWID sont différents, utilisez la commande **RESOLVE CHANNEL** pour rétablir les messages en attente de validation.

Si les deux LUWID sont différents, le destinataire n'a pas validé l'unité de travail que l'expéditeur considère comme en attente de validation. Le côté émetteur doit conserver les messages en attente de validation dans la file d'attente de transmission et les renvoyer. Cette opération est effectuée à l'aide de la commande **RESOLVE CHANNEL** suivante:

```
RESOLVE CHANNEL( name ) ACTION(BACKOUT)
```

IBM i Sous IBM i, vous pouvez utiliser la commande Resolve MQM Channel, **RSVMQMCHL**.

Résultats

Une fois ce processus terminé, le canal n'est plus en attente de validation. La file d'attente de transmission peut désormais être utilisée par un autre canal, si nécessaire.

Référence associée

[DISPLAY CHSTATUS \(statut du canal d'affichage\)](#)

[RESOLVE CHANNEL \(demande à un canal de résoudre les messages en attente de validation\)](#)

Sécurité des messages

Outre les fonctions de récupération standard d'IBM MQ, la gestion des files d'attente réparties garantit que les messages sont distribués correctement à l'aide d'une procédure de point de synchronisation coordonnée entre les deux extrémités du canal de transmission de messages. Si cette procédure détecte une erreur, elle ferme le canal afin que vous puissiez examiner le problème et conserve les messages en toute sécurité dans la file d'attente de transmission jusqu'à ce que le canal soit redémarré.

La procédure de point de synchronisation présente un avantage supplémentaire dans la mesure où elle tente de récupérer une situation *en attente de validation* lorsque le canal démarre. (*En attente de validation* est le statut d'une unité de récupération pour laquelle un point de synchronisation a été demandé, mais le résultat de la demande n'est pas encore connu.) Les deux fonctions suivantes sont également associées à cette fonction:

1. Résolution avec validation ou annulation
2. Réinitialiser le numéro de séquence

L'utilisation de ces fonctions ne se produit que dans des circonstances exceptionnelles, car le canal se rétablit automatiquement dans la plupart des cas.

Messages rapides et non persistants

L'attribut de canal de débit de messages non persistants (NPM SPEED) peut être utilisé pour indiquer que les messages non persistants sur le canal doivent être distribués plus rapidement. Pour plus d'informations sur cet attribut, voir [Vitesse de messages non persistants \(NPM SPEED\)](#).

Si un canal s'arrête alors que des messages rapides et non persistants sont en transit, les messages risquent d'être perdus et il revient à l'application d'organiser leur reprise si nécessaire.

Si le canal récepteur ne peut pas placer le message dans sa file d'attente de destination, il est placé dans la file d'attente des messages non livrés, si elle a été définie. Si ce n'est pas le cas, le message est supprimé.

Remarque : Si l'autre extrémité du canal ne prend pas en charge l'option, le canal s'exécute à une vitesse normale.

Messages non distribués

Pour plus d'informations sur ce qui se passe lorsqu'un message ne peut pas être distribué, voir [«Que se passe-t-il lorsqu'un message ne peut pas être distribué?»](#), à la page 259.

Que se passe-t-il lorsqu'un message ne peut pas être distribué?

Lorsqu'un message ne peut pas être distribué, l'agent MCA peut le traiter de plusieurs manières. Il peut réessayer, renvoyer à l'expéditeur ou le placer dans la file d'attente des messages non livrés.

La [Figure 23](#), à la page 259 montre le traitement qui se produit lorsqu'un agent MCA ne parvient pas à insérer un message dans la file d'attente de destination. (Les options affichées ne s'appliquent pas sur toutes les plateformes.)

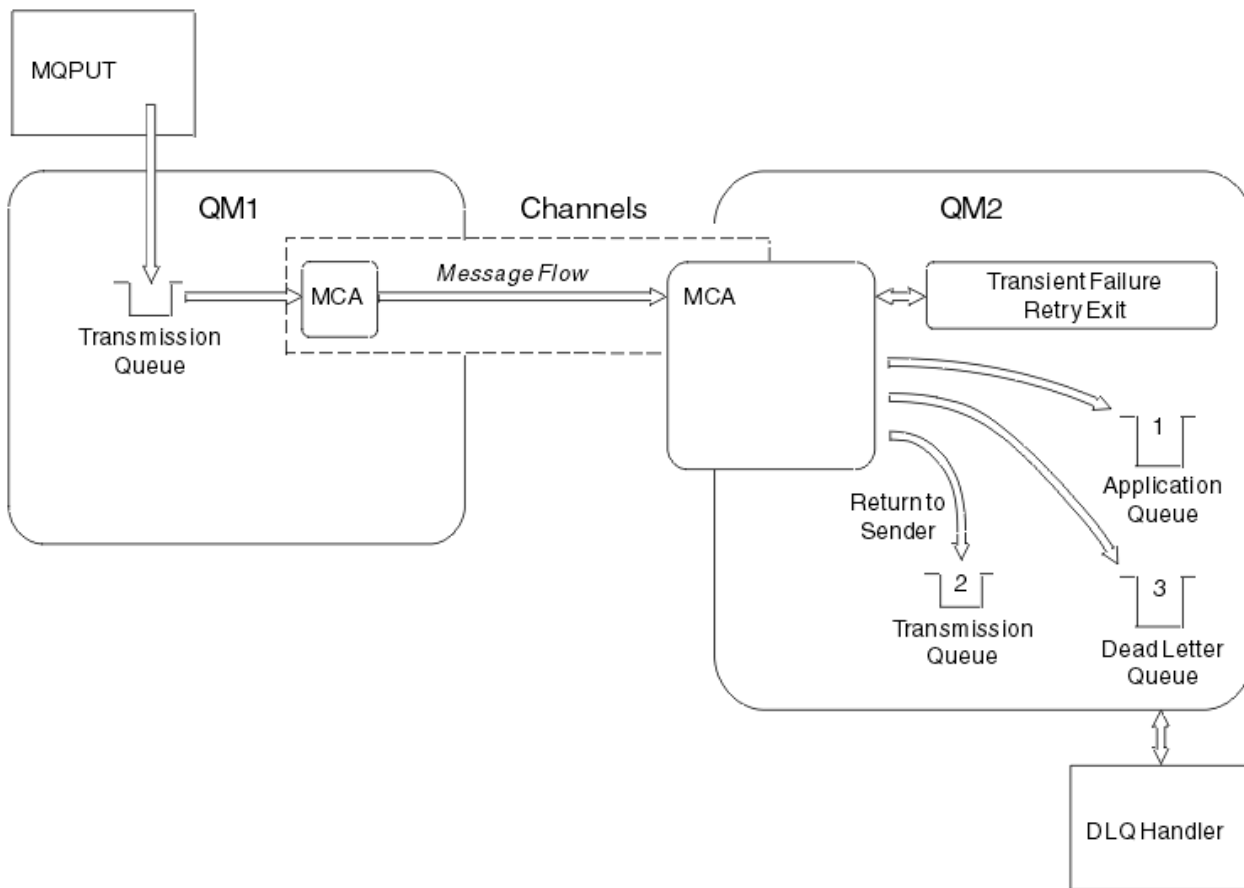


Figure 23. Que se passe-t-il lorsqu'un message ne peut pas être distribué?

Comme illustré dans la figure, l'agent MCA peut effectuer plusieurs opérations avec un message qu'il ne peut pas distribuer. L'action effectuée est déterminée par les options spécifiées lorsque le canal est défini et par les options de rapport MQPUT pour le message.

1. relance de message

Si l'agent MCA ne parvient pas à insérer un message dans la file d'attente cible pour une raison qui peut être transitoire (par exemple, parce que la file d'attente est saturée), l'agent MCA peut attendre et relancer l'opération ultérieurement. Vous pouvez déterminer si l'agent MCA attend, pendant combien de temps et combien de fois il tente de le faire.

- Vous pouvez spécifier un délai et un intervalle de relance de message pour les erreurs MQPUT lorsque vous définissez votre canal. Si le message ne peut pas être inséré dans la file d'attente de destination car la file d'attente est saturée ou est interdite pour les insertions, l'agent MCA tente d'exécuter l'opération le nombre de fois indiqué, à l'intervalle de temps indiqué.
- Vous pouvez écrire votre propre exit de relance de message. L'exit vous permet d'indiquer dans quelles conditions vous souhaitez que l'agent MCA tente à nouveau l'opération MQPUT ou MQOPEN. Indiquez le nom de l'exit lorsque vous définissez le canal.

2. retour à l'expéditeur

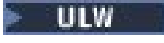

Si la relance de message a échoué ou qu'un autre type d'erreur a été détecté, l'agent MCA peut renvoyer le message à l'émetteur. Pour activer le retour à l'expéditeur, vous devez spécifier les options suivantes dans le descripteur de message lorsque vous placez le message dans la file d'attente d'origine:

- Option de rapport MQRO_EXCEPTION_WITH_FULL_DATA
- Option de rapport MQRO_DISCARD_MSG
- Nom de la file d'attente de réponse et du gestionnaire de files d'attente de réponse

Si l'agent MCA ne parvient pas à insérer le message dans la file d'attente de destination, il génère un rapport d'exception contenant le message d'origine et le place dans une file d'attente de transmission à envoyer à la file d'attente de réponse indiquée dans le message d'origine. (Si la file d'attente de réponse se trouve dans le même gestionnaire de files d'attente que l'agent MCA, le message est placé directement dans cette file d'attente et non dans une file d'attente de transmission.)

3. File d'attente de rebut

Si un message ne peut pas être distribué ou renvoyé, il est placé dans la file d'attente des messages non livrés (DLQ). Vous pouvez utiliser le gestionnaire DLQ pour traiter le message. Ce traitement est décrit ici:

-  [Traitement des messages dans une file d'attente de rebut](#)
-  [Utilitaire de gestionnaire de files d'attente de rebut \(CSQUDLQH\)](#)

Si la file d'attente de rebut n'est pas disponible, l'agent MCA émetteur laisse le message dans la file d'attente de transmission et le canal s'arrête. Sur un canal rapide, les messages non persistants qui ne peuvent pas être écrits dans une file d'attente de rebut sont perdus.

Sous IBM WebSphere MQ 7.0, si aucune file d'attente de rebut locale n'est définie, que la file d'attente éloignée n'est pas disponible ou définie et qu'il n'existe pas de file d'attente de rebut éloignée, le canal émetteur passe dans RETRY et les messages sont automatiquement renvoyés dans la file d'attente de transmission.

Référence associée



[Utiliser la file d'attente des messages non livrés \(USEDLQ\)](#)

Canaux de déclenchement

IBM MQ fournit une fonction permettant de démarrer automatiquement une application lorsque certaines conditions d'une file d'attente sont remplies. Cette fonction est appelée déclenchement.

Cette explication est conçue comme une présentation des concepts de déclenchement. Pour une description complète, voir [Démarrage des applications IBM MQ à l'aide de déclencheurs](#).

Pour obtenir des informations spécifiques à la plateforme, voir:

- Pour AIX, Linux, and Windows, voir [«Déclenchement de canaux sur AIX, Linux, and Windows.»](#), à la page 262
-  Pour IBM i, voir [«Déclenchement de canaux dans IBM MQ for IBM i»](#), à la page 263
-  Pour z/OS, voir [«Transmission queues and triggering channels»](#), à la page 1012

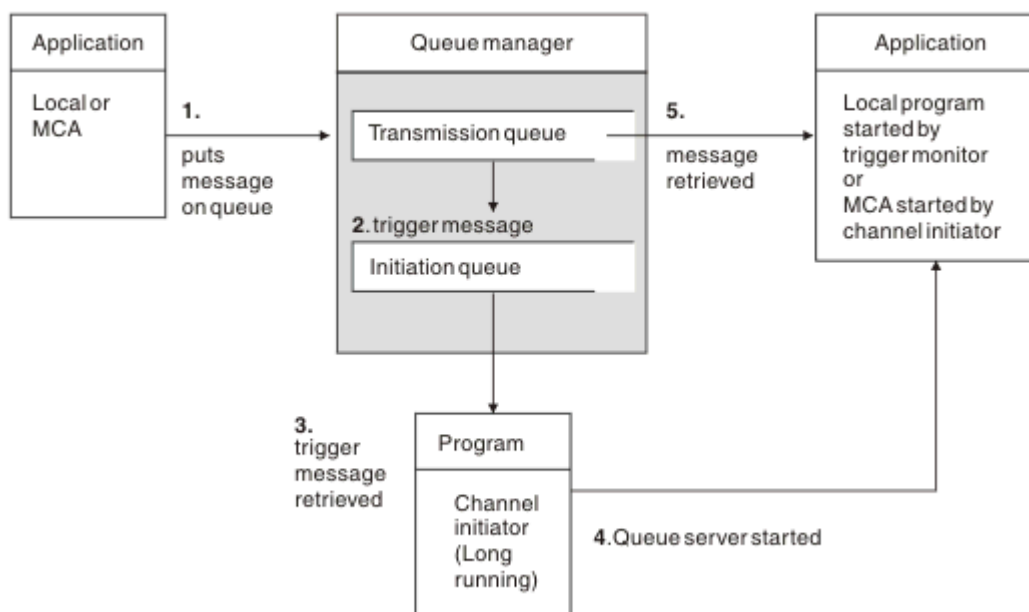



Figure 24. Les concepts de déclenchement


Les objets requis pour le déclenchement sont affichés dans la [Figure 24](#), à la page 261. Il affiche la séquence d'événements suivante:

1. Le gestionnaire de files d'attente local place un message provenant d'une application ou d'un agent MCA dans la file d'attente de transmission.
2. Lorsque les conditions de déclenchement sont remplies, le gestionnaire de files d'attente local place un message de déclenchement dans la file d'attente d'initialisation.
3. Le programme initiateur de canal à exécution longue surveille la file d'attente d'initialisation et extrait les messages lorsqu'ils arrivent.
4. L'initiateur de canal traite les messages de déclenchement en fonction des informations qu'ils contiennent. Ces informations peuvent inclure le nom du canal, auquel cas l'agent MCA correspondant est démarré.
5. L'application locale ou l'agent MCA, ayant été déclenchée, extrait les messages de la file d'attente de transmission.

Pour configurer ce scénario, vous devez:

- Créez la file d'attente de transmission avec le nom de la file d'attente d'initialisation (c'est-à-dire, SYSTEM.CHANNEL.INITQ) dans l'attribut correspondant.
- Vérifiez que la file d'attente d'initialisation (SYSTEM.CHANNEL.INITQ) existe.
- Vérifiez que le programme d'initiation de canal est disponible et en cours d'exécution. Le programme initiateur de canal doit être indiqué avec le nom de la file d'attente d'initialisation dans sa commande de

démarrage.  Sous z/OS, le nom de la file d'attente d'initialisation est fixe et n'est donc pas utilisé dans la commande de démarrage.

- Vous pouvez éventuellement créer la définition de processus pour le déclenchement, si elle n'existe pas, et vous assurer que la zone *UserData* contient le nom du canal qu'elle dessert. Au lieu de créer une définition de processus, vous pouvez spécifier le nom du canal dans l'attribut **TriggerData** de la file d'attente de transmission. IBM MQ pour  IBM i, systèmes AIX, Linux, and Windows , permettent de spécifier un nom de canal vide, auquel cas la première définition de canal disponible avec cette file d'attente de transmission est utilisée.
- Assurez-vous que la définition de la file d'attente de transmission contient le nom de la définition de processus à utiliser (le cas échéant), le nom de la file d'attente d'initialisation et les caractéristiques de déclenchement qui vous semblent les plus appropriées. L'attribut de contrôle de déclencheur permet d'activer ou de ne pas activer le déclenchement, selon les besoins.

Remarque :

1. Le programme initiateur de canal agit en tant que 'moniteur de déclenchement' surveillant la file d'attente d'initialisation utilisée pour démarrer les canaux.
2. Une file d'attente d'initialisation et un processus de déclenchement peuvent être utilisés pour déclencher n'importe quel nombre de canaux.
3. Vous pouvez définir un nombre illimité de files d'attente d'initialisation et de processus de déclenchement.
4. Le type de déclencheur FIRST est recommandé pour éviter d'inonder le système avec des démarrages de canal.

Déclenchement de canaux sur AIX, Linux, and Windows.



Vous pouvez créer une définition de processus dans IBM MQ, en définissant les processus à déclencher. Utilisez la commande MQSC DEFINE PROCESS pour créer une définition de processus nommant le processus à déclencher lorsque des messages arrivent dans une file d'attente de transmission. L'attribut USERDATA de la définition de processus contient le nom du canal servi par la file d'attente de transmission.

Définissez la file d'attente locale (QM4), en spécifiant que les messages de déclenchement doivent être écrits dans la file d'attente d'initialisation (IQ) pour déclencher l'application qui démarre le canal (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ) PROCESS(P1) USAGE(XMITQ)
```

Définissez l'application (processus P1) à démarrer:

```
DEFINE PROCESS(P1) USERDATA(QM3.TO.QM4)
```

Sinon, pour les systèmes IBM MQ for UNIX, Linux et Windows , vous pouvez éliminer la nécessité d'une définition de processus en spécifiant le nom de canal dans l'attribut TRIGDATA de la file d'attente de transmission.

Définissez la file d'attente locale (QM4). Indiquez que les messages de déclenchement doivent être écrits dans la file d'attente d'initialisation par défaut SYSTEM.CHANNEL.INITQ, pour déclencher l'application (processus P1) qui démarre le canal (QM3.TO.QM4):

```
DEFINE QLOCAL(QM4) TRIGGER INITQ(SYSTEM.CHANNEL.INITQ)  
USAGE(XMITQ) TRIGDATA(QM3.TO.QM4)
```

Si vous n'indiquez pas de nom de canal, l'initiateur de canal recherche les fichiers de définition de canal jusqu'à ce qu'il trouve un canal associé à la file d'attente de transmission nommée.

Déclenchement de canaux dans IBM MQ for IBM i

IBM i

Le déclenchement des canaux dans IBM MQ for IBM i est implémenté avec le processus d'initiateur de canal. Un processus d'initialisateur de canal pour la file d'attente d'initialisation SYSTEM.CHANNEL.INITQ est démarré automatiquement avec le gestionnaire de files d'attente, sauf s'il est désactivé en modifiant l'attribut SCHINIT du gestionnaire de files d'attente.

Configurez la file d'attente de transmission pour le canal en spécifiant SYSTEM.CHANNEL.INITQ comme file d'attente d'initialisation et activation du déclenchement pour la file d'attente. L'initiateur de canal démarre le premier canal disponible qui spécifie cette file d'attente de transmission.

```
CRTMQMQ QNAME(MYXMITQ1) QTYPE(*LCL) MQMNAME(MYQMGR)
TRGENBL(*YES) INITQNAME(SYSTEM.CHANNEL.INITQ)
USAGE(*TMQ)
```

Deprecated Vous pouvez démarrer manuellement jusqu'à trois processus initiateurs de canal à l'aide de la commande STRMQMCHLI et spécifier des files d'attente d'initialisation différentes. Vous pouvez également spécifier plusieurs canaux pouvant traiter la file d'attente de transmission et choisir le canal à démarrer. Cette fonction est toujours fournie pour être compatible avec les versions antérieures. Son utilisation est obsolète.

Remarque : Un seul canal à la fois peut traiter une file d'attente de transmission.

```
STRMQMCHLI QNAME(MYINITQ)
```

Configurez la file d'attente de transmission du canal en indiquant TRGENBL (*YES) et, pour choisir le canal à démarrer, indiquez le nom du canal dans la zone TRIGDATA. Exemple :

```
CRTMQMQ QNAME(MYXMITQ2) QTYPE(*LCL) MQMNAME(MYQMGR)
TRGENBL(*YES) INITQNAME(MYINITQ)
USAGE(*TMQ) TRIGDATA(MYCHANNEL)
```

Concepts associés

«Démarrage et arrêt de l'initiateur de canal», à la page 263

Le déclenchement est implémenté à l'aide du processus initiateur de canal.

Tâches associées

«Configuration de la file d'attente répartie», à la page 212

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ, y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Référence associée

Programmes de canal sous AIX, Linux, and Windows

IBM i

Travaux d'intercommunication sous IBM i

IBM i

Etats des canaux sous IBM i

Démarrage et arrêt de l'initiateur de canal

Le déclenchement est implémenté à l'aide du processus initiateur de canal.

Ce processus d'initiateur de canal est démarré à l'aide de la commande MQSC START CHINIT. Sauf si vous utilisez la file d'attente d'initialisation par défaut, indiquez le nom de la file d'attente d'initialisation dans la commande. Par exemple, pour utiliser la commande START CHINIT afin de démarrer la file d'attente IQ pour le gestionnaire de files d'attente par défaut, entrez:

```
START CHINIT INITQ(IQ)
```

Par défaut, un initiateur de canal est démarré automatiquement à l'aide de la file d'attente d'initialisation par défaut, SYSTEM.CHANNEL.INITQ. Si vous souhaitez démarrer tous vos initiateurs de canal manuellement, procédez comme suit:

1. Créez et démarrez le gestionnaire de files d'attente.
2. Modifiez la propriété SCHINIT du gestionnaire de files d'attente en MANUAL
3. Arrêtez et redémarrez le gestionnaire de files d'attente

Dans les systèmes IBM MQ for Multiplatforms , un initiateur de canal est démarré automatiquement. Le nombre d'initiateurs de canal que vous pouvez démarrer est limité. La valeur par défaut et maximale est 3. Vous pouvez le modifier à l'aide de MAXINITIATEURS dans le fichier qm.ini pour les systèmes AIX and Linux et dans le registre pour les systèmes Windows .

Pour plus d'informations sur la commande d'initiateur de canal d'exécution **runmqchi**et sur les autres commandes de contrôle, voir [IBM MQ Commandes de contrôle](#) .

Arrêt de l'initiateur de canal

L'initiateur de canal par défaut est démarré automatiquement lorsque vous démarrez un gestionnaire de files d'attente. Tous les initiateurs de canal sont arrêtés automatiquement lorsqu'un gestionnaire de files d'attente est arrêté.

Fichiers d'initialisation et de configuration

Le traitement des données d'initialisation de canal dépend de votre plateforme IBM MQ .

IBM MQ for z/OS



Dans IBM MQ for z/OS, les informations d'initialisation et de configuration sont spécifiées à l'aide de la commande **ALTER QMGR** MQSC. Si vous placez des commandes **ALTER QMGR** dans le fichier d'entrée d'initialisation CSQINP2 , elles sont traitées chaque fois que le gestionnaire de files d'attente est démarré.

Pour exécuter des commandes MQSC telles que **START LISTENER** chaque fois que vous démarrez l'initiateur de canal, placez-les dans le fichier d'entrée d'initialisation CSQINPX et spécifiez l'instruction de définition de données facultative CSQINPX dans la procédure de tâche démarrée de l'initiateur de canal.

Pour plus d'informations sur CSQINP2 et CSQINPX, voir [Personnalisation des fichiers d'entrée d'initialisation et ALTER QMGR](#).

IBM MQ for Multiplatforms



Dans IBM MQ for Multiplatforms, il existe des fichiers de configuration qui contiennent des informations de configuration de base sur l'installation d' IBM MQ .

Il existe deux fichiers de configuration: l'un s'applique à la machine, l'autre s'applique à un gestionnaire de files d'attente individuel.

Fichier de configuration IBM MQ

Ce fichier contient les informations relatives à tous les gestionnaires de files d'attente sur le système IBM MQ . Le fichier s'appelle `mqc.ini`. Il est décrit dans [«Fichier de configuration IBM MQ , mqs.ini»](#), à la page 98.

Fichier de configuration du gestionnaire de files d'attente

Ce fichier contient les informations de configuration relatives à un gestionnaire de files d'attente particulier. Le fichier est appelé `qm.ini`.

Il est créé lors de la création du gestionnaire de files d'attente et peut contenir des informations de configuration relatives à n'importe quel aspect du gestionnaire de files d'attente. Les informations contenues dans le fichier incluent des détails sur la façon dont la configuration du journal diffère de la configuration par défaut dans le fichier de configuration IBM MQ .

Le fichier de configuration du gestionnaire de files d'attente se trouve à la racine de l'arborescence de répertoires occupée par le gestionnaire de files d'attente. Par exemple, pour les attributs **DefaultPath** , les fichiers de configuration de gestionnaire de files d'attente pour un gestionnaire de files d'attente appelé QMNAME sont les suivants:

Pour les systèmes AIX and Linux :

```
/var/mqm/qmgrs/QMNAME/qm.ini
```

Pour les systèmes Windows :

```
C:\ProgramData\IBM\MQ\qmgrs\QMNAME\qm.ini
```

 Pour IBM i :

```
/QIBM/UserData/mqm/qmgrs/QMNAME/qm.ini
```

Voici un extrait d'un `qm.ini`. Il indique que le programme d'écoute TCP/IP doit être à l'écoute sur le port 2500, que le nombre maximal de canaux en cours est 200 et que le nombre maximal de canaux actifs est 100.

```
TCP:
Port=2500
CHANNELS:
MaxChannels=200
MaxActiveChannels=100
```

Vous pouvez spécifier une plage de ports TCP/IP à utiliser par un canal sortant. L'une des méthodes consiste à utiliser le fichier `qm.ini` pour spécifier le début et la fin d'une plage de valeurs de port. L'exemple suivant illustre un fichier `qm.ini` spécifiant une plage de canaux:

```
TCP:
StrPort=2500
EndPort=3000
CHANNELS:
MaxChannels=200
MaxActiveChannels=100
```

Si vous spécifiez une valeur pour **StrPort** ou **EndPort** , vous devez spécifier une valeur pour les deux. La valeur de **EndPort** doit toujours être supérieure à la valeur de **StrPort**.

Le canal tente d'utiliser chacune des valeurs de port dans la plage spécifiée. Lorsque la connexion aboutit, la valeur de port correspond au port utilisé par le canal.

Pour plus d'informations sur les fichiers `qm.ini` , voir «[Fichiers de configuration du gestionnaire de files d'attente, qm.ini](#)», à la page 111.

Conversion de données pour les messages

Les messages IBM MQ peuvent nécessiter une conversion de données lorsqu'ils sont envoyés entre des files d'attente de différents gestionnaires de files d'attente.

Un message IBM MQ se compose de deux parties:

- Informations de contrôle dans un descripteur de message
- Données d'application

L'une des deux parties peut nécessiter une conversion de données lorsqu'elle est envoyée entre des files d'attente sur des gestionnaires de files d'attente différents. Pour plus d'informations sur la conversion des données d'application, voir [Conversion des données d'application](#).

Écriture de vos propres agents MCA

IBM MQ vous permet d'écrire vos propres programmes MCA (Message Channel Agent) ou d'en installer un à partir d'un fournisseur de logiciels indépendant.

Vous pouvez écrire vos propres programmes MCA pour que IBM MQ interagisse sur votre propre protocole de communication propriétaire ou pour envoyer des messages via un protocole non pris en charge par IBM MQ . (Vous ne pouvez pas écrire votre propre agent MCA pour qu'il interopère avec un agent MCA fourni par IBM MQ à l'autre extrémité.)

Si vous décidez d'utiliser un agent MCA qui n'a pas été fourni par IBM MQ, vous devez prendre en compte les points suivants.

Envoi et réception de messages

Vous devez écrire une application émettrice qui extrait les messages de l'endroit où votre application les insère, par exemple à partir d'une file d'attente de transmission, et les envoie sur un protocole avec lequel vous souhaitez communiquer. Vous devez également écrire une application de réception qui prend les messages de ce protocole et les place dans les files d'attente de destination. Les applications émettrices et réceptrices utilisent les appels de l'interface de file d'attente de messages (MQI), mais pas les interfaces spéciales.

Vous devez vous assurer que les messages ne sont distribués qu'une seule fois. La coordination des points de synchronisation peut être utilisée pour faciliter cette distribution.

Fonction de contrôle de canal

Vous devez fournir vos propres fonctions d'administration pour contrôler les canaux. Vous ne pouvez pas utiliser les fonctions d'administration de canal IBM MQ pour configurer (par exemple, la commande DEFINE CHANNEL) ou surveiller (par exemple, DISPLAY CHSTATUS) vos canaux.

fichier d'initialisation

Vous devez fournir votre propre fichier d'initialisation, si vous en avez besoin.

Conversion des données d'application

Vous souhaitez probablement autoriser la conversion de données pour les messages que vous envoyez à un autre système. Si tel est le cas, utilisez l'option MQGMO_CONVERT sur l'appel MQGET lors de l'extraction de messages à partir de l'emplacement où votre application les insère, par exemple la file d'attente de transmission.

Exits utilisateur

Déterminez si vous avez besoin d'exits utilisateur. Si tel est le cas, vous pouvez utiliser les mêmes définitions d'interface que celles utilisées par IBM MQ .

Déclenchement

Si votre application insère des messages dans une file d'attente de transmission, vous pouvez configurer les attributs de la file d'attente de transmission de sorte que l'agent MCA émetteur soit déclenché lorsque des messages arrivent dans la file d'attente.

Initiateur de canal

Vous devrez peut-être fournir votre propre initiateur de canal.

Autres éléments à prendre en compte pour la gestion de files d'attente réparties

Autres rubriques à prendre en compte lors de la préparation de IBM MQ pour la gestion de files d'attente réparties. Cette rubrique traite des files d'attente de messages non distribuées, des files d'attente en cours d'utilisation, des extensions système et des programmes d'exit utilisateur, ainsi que de l'exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées.

Non distribué-file d'attente de messages

Pour vous assurer que les messages arrivant dans la file d'attente de messages non distribués (également appelée file d'attente de messages non livrés ou file d'attente de messages non livrés) sont traités, créez un programme qui peut être déclenché ou exécuté à intervalles réguliers pour traiter ces messages.

Linux **AIX** Un gestionnaire DLQ est fourni avec IBM MQ sur les systèmes AIX and Linux ; pour plus d'informations, voir [Exemple de gestionnaire DLQ, amqsdlq](#).

IBM i Pour plus d'informations sur IBM MQ for IBM i, voir [Le gestionnaire de files d'attente de rebut IBM MQ for IBM i](#).

Files d'attente en cours d'utilisation

Les agents MCA pour les canaux récepteurs peuvent conserver les files d'attente de destination ouvertes même lorsque les messages ne sont pas transmis. Ainsi, les files d'attente semblent être "en cours d'utilisation".

Nombre maximal de canaux

IBM i Sur IBM MQ for IBM i , vous pouvez spécifier le nombre maximal de canaux autorisés dans votre système et le nombre maximal pouvant être actifs simultanément. Vous spécifiez ces nombres dans le fichier `qm.ini` du répertoire `QIBM/UserData/mqm/qmgrs/nom_gestionnaire_files_attente`. Voir [Sections du fichier de configuration pour la mise en file d'attente répartie](#).

Extensions système et programmes d'exit utilisateur

Une fonction est fournie dans la définition de canal pour permettre l'exécution de programmes supplémentaires à des moments définis lors du traitement des messages. Ces programmes ne sont pas fournis avec IBM MQ, mais peuvent être fournis par chaque installation en fonction des exigences locales.

Pour pouvoir être exécutés, ces programmes d'exit utilisateur doivent avoir des noms prédéfinis et être disponibles sur appel aux programmes de canal. Les noms des programmes d'exit utilisateur sont inclus dans les définitions de canal de transmission de messages.

Il existe une interface de bloc de contrôle définie pour transférer le contrôle à ces programmes, et pour gérer le retour de contrôle à partir de ces programmes.

Les emplacements précis où ces programmes sont appelés, ainsi que les détails des blocs de contrôle et des noms, se trouvent dans [Programmes d'exit de canal pour les canaux de messagerie](#).

Exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées

Si les performances sont une considération importante dans votre environnement et que votre environnement est stable, vous pouvez exécuter vos canaux et programmes d'écoute en tant que confiance, à l'aide de la liaison FASTPATH. Il existe deux facteurs qui influencent le fait que les canaux et les programmes d'écoute s'exécutent comme étant dignes de confiance:

- La variable d'environnement `MQ_CONNECT_TYPE=FASTPATH` ou `MQ_CONNECT_TYPE = STANDARD`. Ceci est sensible à la casse. Si vous indiquez une valeur incorrecte, elle est ignorée.
- `MQIBindType` dans la strophe `Channels` du fichier `qm.ini` ou du fichier de registre. Vous pouvez la définir sur `FASTPATH` ou `STANDARD` et elle n'est pas sensible à la casse. La valeur par défaut est `Standard`.

Vous pouvez utiliser `MQIBindType` en association avec la variable d'environnement pour obtenir l'effet requis comme suit:

MQIBindType	Variable d'environnement	Résultat
STANDARD	NON DEFINI	STANDARD

MQIBindType	Variable d'environnement	Résultat
Fastpath	NON DEFINI	Fastpath
STANDARD	STANDARD	STANDARD
Fastpath	STANDARD	STANDARD
STANDARD	Fastpath	STANDARD
Fastpath	Fastpath	Fastpath
STANDARD	CLIENT	CLIENT
Fastpath	CLIENT	STANDARD
STANDARD	LOCAL	STANDARD
Fastpath	LOCAL	STANDARD

En résumé, il n'y a que deux façons de faire en sorte que les canaux et les programmes d'écoute soient considérés comme dignes de confiance:

1. En spécifiant MQIBindType= FASTPATH dans `qm.ini` ou dans le registre et en ne spécifiant pas la variable d'environnement.
2. En spécifiant MQIBindType= FASTPATH dans `qm.ini` ou dans le registre et en définissant la variable d'environnement sur FASTPATH.

Pensez à exécuter des programmes d'écoute comme étant dignes de confiance, car les programmes d'écoute sont des processus stables. Considérez les canaux en cours d'exécution comme dignes de confiance, sauf si vous utilisez des exits de canal instables ou la commande STOP CHANNEL MODE (TERMINATE).

ALW

Surveillance et contrôle des canaux sur AIX, Linux, and Windows

Pour DQM, vous devez créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Vous pouvez contrôler les canaux à l'aide de commandes, de programmes, de IBM MQ Explorer, de fichiers pour les définitions de canal et d'une zone de stockage pour les informations de synchronisation.

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser les types de commande suivants pour contrôler les canaux:

Les commandes IBM MQ (MQSC)

Vous pouvez utiliser le MQSC en tant que commandes uniques dans une session MQSC dans les systèmes AIX, Linux, and Windows . Pour émettre des commandes plus complexes ou multiples, le MQSC peut être intégré dans un fichier que vous exécutez ensuite à partir de la ligne de commande. Pour plus de détails, voir [Commandes MQSC](#). Cette section fournit des exemples simples d'utilisation de MQSC pour la mise en file d'attente répartie.

Les commandes de canal sont un sous-ensemble des commandes IBM MQ (MQSC). Vous utilisez MQSC et les commandes de contrôle pour:

- Créer, copier, afficher, modifier et supprimer des définitions de canal
- Démarrer et arrêter les canaux, exécuter une commande ping, réinitialiser les numéros de séquence des canaux et résoudre les messages en attente de validation lorsque les liens ne peuvent pas être rétablis
- Afficher les informations de statut sur les canaux

Commandes de contrôle

Vous pouvez également exécuter des *commandes de contrôle* sur la ligne de commande pour certaines de ces fonctions. Pour plus de détails, voir [Administration d' IBM MQ for Multiplatforms à l'aide de commandes de contrôle](#).

Commandes de format de commande programmable

Pour plus de détails, voir [Commandes PCF](#).

IBM MQ Explorer

Sur les systèmes Linux et Windows , vous pouvez utiliser IBM MQ Explorer. Il s'agit d'une interface d'administration graphique permettant d'effectuer des tâches d'administration au lieu d'utiliser des commandes de contrôle ou des commandes MQSC. Les définitions de canal sont conservées en tant qu'objets de gestionnaire de files d'attente.

Chaque gestionnaire de files d'attente possède un composant DQM permettant de contrôler les interconnexions aux gestionnaires de files d'attente éloignées compatibles. Une zone de stockage contient des numéros de séquence et des identificateurs d' *unité de travail logique (LUW)* . Ils sont utilisés à des fins de synchronisation de canal.

Pour obtenir la liste des fonctions disponibles lors de la configuration et du contrôle des canaux de transmission de messages à l'aide des différents types de commande, voir [Tableau 22, à la page 270](#).

Procédure

- [«Fonctions requises pour la configuration et le contrôle des canaux», à la page 269](#)
- [«Initiation aux objets», à la page 272](#)
- [«Configuration de la communication sur Windows», à la page 279](#)
- [«Configuration de la communication sur AIX and Linux», à la page 287](#)


Tâches associées

[«Surveillance et contrôle des canaux sur IBM i», à la page 293](#)

Utilisez les commandes et les panneaux DQM pour créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Chaque gestionnaire de files d'attente dispose d'un programme DQM permettant de contrôler les interconnexions aux gestionnaires de files d'attente éloignées compatibles.

Référence associée

 [Programmes de canal sous AIX, Linux, and Windows](#)

 [Exemple de planification de canal de transmission de messages pour AIX, Linux, and Windows](#)

[Exemple d'informations de configuration](#)

[Attributs de canal](#)

Fonctions requises pour la configuration et le contrôle des canaux

Un certain nombre de fonctions IBM MQ peuvent être nécessaires pour configurer et contrôler les canaux. Les fonctions de canal sont expliquées dans cette rubrique.

Vous pouvez créer une définition de canal à l'aide des valeurs par défaut fournies par IBM MQ, en spécifiant le nom du canal, le type de canal que vous créez, la méthode de communication à utiliser, le nom de la file d'attente de transmission et le nom de la connexion.

Le nom de canal doit être identique aux deux extrémités du canal et unique au sein du réseau. Toutefois, vous devez limiter les caractères utilisés à ceux qui sont valides pour les noms d'objet IBM MQ .

Pour d'autres fonctions liées aux canaux, voir les rubriques suivantes:

- [«Initiation aux objets», à la page 272](#)
- [«Création d'objets associés», à la page 272](#)

- «Création d'objets par défaut», à la page 273
- «Création d'un canal», à la page 273
- «Affichage d'un canal», à la page 274
- «Affichage de l'état du canal», à la page 274
- «Vérification des liens à l'aide de Ping», à la page 275
- «Démarrage d'un canal», à la page 275
- «Arrêt d'un canal», à la page 277
- «Changement de nom d'un canal», à la page 277
- «Réinitialisation d'un canal», à la page 278
- «Résolution des messages en attente de validation sur un canal», à la page 278

Le Tableau 22, à la page 270 présente la liste complète des fonctions IBM MQ dont vous pourriez avoir besoin.

<i>Tableau 22. Fonctions requises dans les systèmes AIX, Linux, and Windows</i>			
Fonction	Commandes de contrôle	MQSC	IBM MQ Explorer équivalent?
Fonctions du gestionnaire de files d'attente			
Modifier un gestionnaire de files d'attente		ALTER QMGR	Oui
De la création d'un gestionnaire de files d'attente	crtmqm		Oui
Supprimer le gestionnaire de files d'attente	dlmqm		Oui
Afficher le gestionnaire de files d'attente		DISPLAY QMGR	Oui
Arrêter le gestionnaire de files d'attente	endmqm		Oui
Envoyer une commande Ping à un gestionnaire de files d'attente		Commande PING QMGR	Non
Démarrer le gestionnaire de files d'attente	strmqm		Oui
Fonctions du serveur de commandes			
Serveur de commandes d'affichage	dspmqcsv		Non
Arrêter le serveur de commandes	endmqcsv		Non
Démarrer un serveur de commandes	strmqcsv		Non
Fonctions de file d'attente			
Modifier une file d'attente		ALTER QALIAS ALTER QLOCAL ALTER QMODEL ALTER QREMOTE Voir ALTER queues .	Oui
Effacer la file d'attente		CLEAR QLOCAL	Oui
Créer une file d'attente		DEFINE QALIAS DEFINE QLOCAL DEFINE QMODEL DEFINE QREMOTE Voir DEFINE queues .	Oui

Tableau 22. Fonctions requises dans les systèmes AIX, Linux, and Windows (suite)




Fonction	Commandes de contrôle	MQSC	IBM MQ Explorer équivalent?
Supprimer la file d'attente		DELETE QALIAS DELETE QLOCAL DELETE QMODEL DELETE QREMOTE Voir DELETE queues .	Oui
DISPLAY QUEUE		DISPLAY QUEUE	Oui
fonctions de processus			
Changement de processus		ALTER PROCESS	Oui
Créer un processus		DEFINE PROCESSUS	Oui
Supprimer le processus		Supprimer un processus	Oui
Processus d'affichage		PROCESSUS d'affichage	Oui
Fonctions de canal			
Modifier un canal		ALTER CHANNEL	Oui
Créer un canal		De la définition d'un canal	Oui
Supprimer le canal		Supprimer le canal	Oui
Canal d'affichage		CANAL D'AFFICHAGE	Oui
Afficher le statut du canal		DISPLAY CHSTATUS	Oui
Canal d'extrémité		Arrêter le canal	Oui
Envoyer une commande PING à un canal		Envoyer une commande PING à un canal	Oui
Réinitialiser le canal		Réinitialisation du canal	Oui
Résolution du canal		Résolution du canal	Oui
Canal d'exécution	runmqchl	Démarrer un canal	Oui
 Exécuter l'initiateur de canal	runmqchi	START CHINIT	Non
Exécuter le programme d'écoute ¹	runmqlsr	Démarrer le programme d'écoute	Non

Tableau 22. Fonctions requises dans les systèmes AIX, Linux, and Windows (suite)

Fonction	Commandes de contrôle	MQSC	IBM MQ Explorer équivalent?
Arrêter le programme d'écoute	endmqsr, sur les plateformes suivantes uniquement: <ul style="list-style-type: none"> •  AIX •  Windows Systèmes Windows		Non

Remarque :

1. Un programme d'écoute peut être démarré automatiquement lorsque le gestionnaire de files d'attente démarre.

 **Initiation aux objets**

Les canaux doivent être définis et leurs objets associés doivent exister et être disponibles pour être utilisés avant qu'un canal puisse être démarré. Cette section vous montre comment procéder.

Utilisez les commandes IBM MQ (MQSC) ou IBM MQ Explorer pour:

1. Définition des canaux de transmission de messages et des objets associés
2. Surveiller et contrôler les canaux de transmission de messages

Les objets associés que vous devrez peut-être définir sont les suivants:

- Files d'attente de transmission
- Définition de files d'attente éloignées
- Définition des alias de gestionnaire de files d'attente
- Définition des alias de files d'attente de réponse
- Files d'attente locales de réponse
- Processus de déclenchement (MCA)
- Définitions de canal de transmission de messages

La liaison de communication particulière pour chaque canal doit être définie et disponible pour qu'un canal puisse être exécuté. Pour une description de la manière dont les liaisons LU 6.2, TCP/IP, NetBIOS, SPX et DECnet sont définies, consultez le guide de communication spécifique à votre installation. Voir aussi [Exemple d'informations de configuration](#).

Pour plus d'informations sur la création et l'utilisation d'objets, voir les sous-rubriques suivantes:

 **Création d'objets associés**

MQSC est utilisé pour créer des objets associés.

Utilisez MQSC pour créer la file d'attente et les objets alias: files d'attente de transmission, définitions de file d'attente éloignée, définitions d'alias de gestionnaire de files d'attente, définitions d'alias de file d'attente de réponse et files d'attente locales de réponse.

Créez également les définitions des processus de déclenchement (MCM) d'une manière similaire.

Pour un exemple montrant comment créer tous les objets requis, voir [Exemple de planification de canal de message pour AIX, Linux, and Windows](#).

Création d'objets par défaut

Les objets par défaut sont créés automatiquement lorsqu'un gestionnaire de files d'attente est créé. Ces objets sont des files d'attente, des canaux, une définition de processus et des files d'attente d'administration. Une fois les objets par défaut créés, vous pouvez les remplacer à tout moment en exécutant la commande `strmqm` avec l'option `-c`.

Lorsque vous utilisez la commande `crtmqm` pour créer un gestionnaire de files d'attente, la commande lance également un programme pour créer un ensemble d'objets par défaut.

1. Chaque objet par défaut est créé à son tour. Le programme tient compte du nombre d'objets définis avec succès, du nombre d'objets existants et remplacés et du nombre de tentatives infructueuses.
2. Le programme affiche les résultats pour vous et, si des erreurs se sont produites, vous dirige vers le journal des erreurs approprié pour plus de détails.

Une fois l'exécution du programme terminée, vous pouvez utiliser la commande `strmqm` pour démarrer le gestionnaire de files d'attente.

Pour plus d'informations sur les commandes `crtmqm` et `strmqm`, voir [Administration d' IBM MQ for Multiplatforms à l'aide de commandes de contrôle](#) .

Modification des objets par défaut

Lorsque vous spécifiez l'option `-c`, le gestionnaire de files d'attente est démarré temporairement lors de la création des objets, puis il est à nouveau arrêté. L'émission de la commande `strmqm` avec l'option `-c` actualise les objets système existants avec les valeurs par défaut (par exemple, l'attribut `MCAUSER` d'une définition de canal est mis à blanc). Vous devez utiliser à nouveau la commande `strmqm`, sans l'option `-c`, si vous souhaitez démarrer le gestionnaire de files d'attente.

Si vous souhaitez modifier les objets par défaut, vous pouvez créer votre propre version de l'ancien fichier `amqscoma.tst` et l'éditer.

Création d'un canal

Créez deux définitions de canal, une à chaque extrémité de la connexion. Vous créez la première définition de canal au niveau du premier gestionnaire de files d'attente. Vous créez ensuite la deuxième définition de canal au niveau du deuxième gestionnaire de files d'attente, à l'autre extrémité du lien.

Les deux extrémités doivent être définies à l'aide du même nom de canal. Les deux extrémités doivent avoir des types de canal compatibles, par exemple: émetteur et récepteur.

Pour créer une définition de canal pour une extrémité du lien, utilisez la commande `MQSC DEFINE CHANNEL`. Incluez le nom du canal, le type de canal pour cette extrémité de la connexion, un nom de connexion, une description (si nécessaire), le nom de la file d'attente de transmission (si nécessaire) et le protocole de transmission. Incluez également tous les autres attributs dont vous souhaitez qu'ils soient différents des valeurs par défaut du système pour le type de canal requis, à l'aide des informations que vous avez collectées précédemment.

Vous pouvez déterminer les valeurs des attributs de canal dans la section [Attributs de canal](#).

Remarque : Il est recommandé de nommer tous les canaux de votre réseau de manière unique. L'inclusion des noms de gestionnaire de files d'attente source et cible dans le nom de canal est un bon moyen d'y parvenir.

Exemple de création de canal

```
DEFINE CHANNEL(QM1.TO.QM2) CHLTYPE(SDR) +
DESCR('Sender channel to QM2') +
CONNAME(QM2) TRPTYPE(TCP) XMITQ(QM2) CONVERT(YES)
```

Dans tous les exemples de `MQSC`, la commande est affichée telle qu'elle apparaît dans un fichier de commandes et telle qu'elle est saisie dans AIX, Linux, and Windows. Les deux méthodes sont identiques, sauf que pour émettre une commande de manière interactive, vous devez d'abord démarrer une session

MQSC. Entrez `runmqsc` pour le gestionnaire de files d'attente par défaut ou `runmqsc qmname` où `qmname` est le nom du gestionnaire de files d'attente requis. Entrez ensuite un nombre quelconque de commandes, comme indiqué dans les exemples.

Pour la portabilité, limitez la longueur de ligne de vos commandes à 72 caractères. Utilisez le caractère de concaténation `+`, comme indiqué pour continuer sur plusieurs lignes:

- **Windows** Sous Windows, utilisez Ctrl-z pour terminer l'entrée sur la ligne de commande.
- **Linux** **AIX** Sous AIX and Linux, utilisez Ctrl-d.
- Vous pouvez également utiliser la commande **end** sous AIX, Linux, and Windows.

ALW *Affichage d'un canal*

Utilisez la commande MQSC `DISPLAY CHANNEL` pour afficher les attributs d'un canal.

Le paramètre `ALL` de la commande `DISPLAY CHANNEL` est pris par défaut si aucun attribut spécifique n'est demandé et que le nom de canal indiqué n'est pas générique.

Les attributs sont décrits dans [Attributs de canal](#).

Exemples de canal d'affichage

```
DISPLAY CHANNEL(QM1.TO.QM2) TRPTYPE, CONVERT
DISPLAY CHANNEL(QM1.TO.*) TRPTYPE, CONVERT
DISPLAY CHANNEL(*) TRPTYPE, CONVERT
DISPLAY CHANNEL(QM1.TO.QMR34) ALL
```

ALW *Affichage de l'état du canal*

Utilisez la commande MQSC `DISPLAY CHSTATUS`, en indiquant le nom du canal et si vous souhaitez connaître le statut en cours des canaux ou le statut des informations sauvegardées.

`DISPLAY CHSTATUS` s'applique à tous les canaux de transmission de messages. Elle ne s'applique pas aux canaux MQI autres que les canaux de connexion serveur.

Les informations affichées sont les suivantes:

- Nom du canal
- Nom de la connexion de communication
- Statut en attente de validation du canal (le cas échéant)
- Dernier numéro de séquence
- Nom de la file d'attente de transmission (le cas échéant)
- Identificateur en attente de validation (le cas échéant)
- Dernier numéro de séquence validé
- Identificateur d'unité d'oeuvre logique
- ID processus
- **Windows** ID d'unité d'exécution (Windows uniquement)

Exemples d'affichage de statut de canal

```
DISPLAY CHSTATUS(*) CURRENT
DISPLAY CHSTATUS(QM1.TO.*) SAVED
```

L'état sauvegardé ne s'applique pas tant qu'au moins un lot de messages n'a pas été transmis sur le canal. L'état est également sauvegardé lorsqu'un canal est arrêté (à l'aide de la commande STOP CHL) et lorsque le gestionnaire de files d'attente est arrêté.

ALW *Vérification des liens à l'aide de Ping*

Utilisez la commande MQSC **PING CHANNEL** pour échanger un message de données fixe avec l'extrémité distante.

La commande PING permet au superviseur du système de s'assurer que le lien est disponible et qu'il fonctionne.

La commande ping n'implique pas l'utilisation de files d'attente de transmission et de files d'attente cible. Il utilise des définitions de canal, la liaison de communication associée et la configuration du réseau. Elle ne peut être utilisée que si le canal n'est pas actuellement actif.

Il est disponible uniquement à partir des canaux émetteur, serveur et émetteur de cluster. Le canal correspondant est démarré à l'extrémité de la liaison et effectue la négociation des paramètres de démarrage. Les erreurs sont notifiées normalement.

Le résultat de l'échange de messages est présenté sous la forme `Ping complete` ou d'un message d'erreur.

Commande ping avec LU 6.2

Lorsque Ping est appelé, par défaut, aucun ID utilisateur ou mot de passe n'est transmis à l'extrémité réceptrice. Si un ID utilisateur et un mot de passe sont requis, ils peuvent être créés à l'extrémité initiatrice dans la définition de canal. Si un mot de passe est entré dans la définition de canal, il est chiffré par IBM MQ avant d'être sauvegardé. Il est ensuite déchiffré avant de traverser la conversation.

Tâches associées

[Utilisation de Ping pour tester les communications](#)

[Test Ping d'un canal afin de vérifier une connexion](#)

Référence associée

[PING CHANNEL \(test de la réponse du canal\)](#)

ALW *Démarrage d'un canal*

Utilisez la commande MQSC **START CHANNEL** pour les canaux émetteur, serveur et demandeur. Pour que les applications puissent échanger des messages, vous devez démarrer un programme d'écoute pour les connexions entrantes.

START CHANNEL n'est pas nécessaire lorsqu'un canal a été configuré avec le déclenchement du gestionnaire de files d'attente.

Une fois démarré, l'agent MCA émetteur lit les définitions de canal et ouvre la file d'attente de transmission. Une séquence de démarrage de canal est émise, qui démarre à distance l'agent MCA correspondant du canal récepteur ou serveur. Une fois qu'ils ont été démarrés, les processus de l'expéditeur et du serveur attendent les messages arrivant dans la file d'attente de transmission et les transmettent à mesure qu'ils arrivent.

Lorsque vous utilisez le déclenchement ou l'exécution de canaux en tant qu'unités d'exécution, vérifiez que l'initiateur de canal est disponible pour surveiller la file d'attente d'initialisation. L'initiateur de canal est démarré par défaut dans le cadre du gestionnaire de files d'attente.

Toutefois, TCP et LU 6.2 fournissent d'autres fonctionnalités:

- ▶ **Linux** ▶ **AIX** Pour TCP sur AIX and Linux, `inetd` peut être configuré pour démarrer un canal. `inetd` est démarré en tant que processus distinct.
- ▶ **Linux** ▶ **AIX** Pour l'unité logique 6.2 dans AIX and Linux, configurez votre produit SNA pour démarrer le processus de réponse de l'unité logique 6.2 .

- **Windows** Pour l'unité logique 6.2 dans Windows, à l'aide de SNA Server, vous pouvez utiliser TpStart (un utilitaire fourni avec SNA Server) pour démarrer un canal. TpStart est démarré en tant que processus distinct.

L'utilisation de l'option Démarrer entraîne toujours la resynchronisation du canal, si nécessaire.

Pour que le démarrage aboutisse:

- Les définitions de canal, locales et distantes, doivent exister. S'il n'existe pas de définition de canal appropriée pour un canal récepteur ou de connexion serveur, un canal par défaut est créé automatiquement si le canal est défini automatiquement. Voir [Programme d'exit de définition automatique de canal](#).
- La file d'attente de transmission doit exister et ne doit pas être utilisée par d'autres canaux.
- Les agents MCA, locaux et distants, doivent exister.
- La liaison de communication doit être disponible.
- Les gestionnaires de files d'attente doivent être en cours d'exécution, locaux et distants.
- Le canal de transmission de messages ne doit pas être déjà en cours d'exécution.

Un message est renvoyé à l'écran pour confirmer que la demande de démarrage d'un canal a été acceptée. Pour confirmer que la commande de démarrage a abouti, consultez le journal des erreurs ou utilisez DISPLAY CHSTATUS. Les journaux d'erreurs sont les suivants:

Windows Windows

MQ_DATA_PATH\mqgrs\qmname\errors\AMQERR01.LOG (pour chaque gestionnaire de files d'attente appelé qmname)

MQ_DATA_PATH\mqgrs\@SYSTEM\errors\AMQERR01.LOG (pour les erreurs générales)

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

Remarque : Sous Windows, vous obtenez toujours un message dans le journal des événements d'application des systèmes Windows .

Linux AIX AIX and Linux

/var/mqm/qmgrs/qmname/errors/AMQERR01.LOG (pour chaque gestionnaire de files d'attente appelé qmname)

/var/mqm/qmgrs/@SYSTEM/errors/AMQERR01.LOG (pour les erreurs générales)

Sous AIX, Linux, and Windows, utilisez la commande **runmqclsr** pour démarrer le processus du programme d'écoute IBM MQ . Par défaut, toute demande entrante de connexion de canal entraîne le processus du programme d'écoute à démarrer les agents MCA en tant qu'unités d'exécution du processus amqrmpa.

```
runmqclsr -t tcp -m QM2
```

Pour les connexions sortantes, vous devez démarrer le canal de l'une des trois manières suivantes:

1. Utilisez la commande MQSC START CHANNEL, en spécifiant le nom du canal, pour démarrer le canal en tant que processus ou unité d'exécution, en fonction du paramètre MCATYPE. (Si les canaux sont démarrés en tant qu'unités d'exécution, il s'agit des unités d'exécution d'un initiateur de canal.)

```
START CHANNEL(QM1.TO.QM2)
```

2. Utilisez la commande de contrôle runmqchl pour démarrer le canal en tant que processus.

```
runmqchl -c QM1.TO.QM2 -m QM1
```

3. Utilisez l'initiateur de canal pour déclencher le canal.

ALW Arrêt d'un canal

Utilisez la commande MQSC STOP CHANNEL pour demander au canal d'arrêter l'activité. Le canal ne démarre pas un nouveau lot de messages tant que l'opérateur ne redémarre pas le canal.

Pour plus d'informations sur le redémarrage des canaux arrêtés, voir «[Redémarrage des canaux arrêtés](#)», à la page 256.

Cette commande peut être émise vers un canal de tout type, à l'exception de MQCHT_CLNTCONN.

Vous pouvez sélectionner le type d'arrêt dont vous avez besoin:

Exemple d'arrêt de la mise au repos

```
STOP CHANNEL(QM1.TO.QM2) MODE(QUIESCE)
```

Cette commande demande au canal de se fermer de manière ordonnée. Le lot de messages en cours est terminé et la procédure de point de synchronisation est effectuée avec l'autre extrémité du canal. Si le canal est inactif, cette commande ne met pas fin à un canal récepteur.

Exemple d'arrêt forcé

```
STOP CHANNEL(QM1.TO.QM2) MODE(FORCE)
```

Cette option arrête le canal immédiatement, mais ne met pas fin à l'unité d'exécution ou au processus du canal. Le canal ne termine pas le traitement du lot de messages en cours et peut donc laisser le canal en attente de validation. En règle générale, envisagez d'utiliser l'option d'arrêt de la mise au repos.

Exemple d'arrêt et d'arrêt

```
STOP CHANNEL(QM1.TO.QM2) MODE(TERMINATE)
```

Cette option arrête le canal immédiatement et arrête l'unité d'exécution ou le processus du canal.

Exemple d'arrêt (quiesce) arrêté

```
STOP CHANNEL(QM1.TO.QM2) STATUS(STOPPED)
```

Cette commande ne spécifiant pas de MODE, la valeur par défaut est MODE (QUIESCE). Il demande que le canal soit arrêté afin qu'il ne puisse pas être redémarré automatiquement mais qu'il soit démarré manuellement.

Exemple d'arrêt (mise au repos) inactif

```
STOP CHANNEL(QM1.TO.QM2) STATUS(INACTIVE)
```

Cette commande ne spécifiant pas de MODE, la valeur par défaut est MODE (QUIESCE). Il demande que le canal soit rendu inactif afin qu'il redémarre automatiquement lorsque cela est nécessaire.

ALW Changement de nom d'un canal

Utilisez MQSC pour renommer un canal de transmission de messages.

Utilisez MQSC pour effectuer les étapes suivantes:

1. Utilisez STOP CHANNEL pour arrêter le canal.
2. Utilisez DEFINE CHANNEL pour créer une définition de canal en double avec le nouveau nom.

3. Utilisez DISPLAY CHANNEL pour vérifier qu'il a été créé correctement.
4. Utilisez DELETE CHANNEL pour supprimer la définition de canal d'origine.

Si vous décidez de renommer un canal de transmission de messages, n'oubliez pas qu'un canal possède deux définitions de canal, une à chaque extrémité. Veillez à renommer le canal aux deux extrémités en même temps.

Réinitialisation d'un canal

Utilisez la commande MQSC RESET CHANNEL pour modifier le numéro de séquence du message.

La commande RESET CHANNEL est disponible pour tous les canaux de transmission de messages, mais pas pour les canaux MQI (connexion client ou connexion serveur). Le premier message démarre la nouvelle séquence lors du prochain démarrage du canal.

Si la commande est émise sur un canal émetteur ou serveur, elle informe l'autre partie de la modification lorsque le canal est redémarré.

Concepts associés

[«Initiation aux objets», à la page 272](#)

Les canaux doivent être définis et leurs objets associés doivent exister et être disponibles pour être utilisés avant qu'un canal puisse être démarré. Cette section vous montre comment procéder.

[«Fonction de contrôle de canal», à la page 244](#)

La fonction de contrôle de canal fournit des fonctions permettant de définir, de surveiller et de contrôler des canaux.

Tâches associées

[«Configuration de la file d'attente répartie», à la page 212](#)

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ, y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Référence associée

[Réinitialisation du canal](#)

Résolution des messages en attente de validation sur un canal

Utilisez la commande MQSC RESOLVE CHANNEL lorsque des messages sont mis en attente de validation par un émetteur ou un serveur. Par exemple, parce qu'une extrémité du lien s'est arrêtée et qu'il n'y a aucune possibilité de récupération.

La commande [RESOLVE CHANNEL](#) accepte l'un des deux paramètres suivants: BACKOUT ou COMMIT. L'annulation restaure les messages dans la file d'attente de transmission, tandis que la validation les supprime.

Le programme de canal ne tente pas d'établir une session avec un partenaire. A la place, il détermine l'identificateur d'unité d'oeuvre logique (LUWID) qui représente les messages en attente de validation. Il émet ensuite, comme demandé, soit:

- BACKOUT pour restaurer les messages dans la file d'attente de transmission ; ou
- COMMIT pour supprimer les messages de la file d'attente de transmission.

Pour que la résolution aboutisse:

- Le canal doit être inactif
- Le canal doit être en attente de validation
- Le type de canal doit être émetteur, serveur ou émetteur de cluster
- Une définition de canal locale doit exister
- Le gestionnaire de files d'attente local doit être en cours d'exécution

Concepts associés

[«Initiation aux objets», à la page 272](#)

Les canaux doivent être définis et leurs objets associés doivent exister et être disponibles pour être utilisés avant qu'un canal puisse être démarré. Cette section vous montre comment procéder.

«Fonction de contrôle de canal», à la page 244

La fonction de contrôle de canal fournit des fonctions permettant de définir, de surveiller et de contrôler des canaux.

Tâches associées

«Configuration de la file d'attente répartie», à la page 212

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ, y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Référence associée


[Résolution du canal](#)

Windows Configuration de la communication sur Windows

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour que cette opération aboutisse, la connexion doit être définie et disponible. Cette section explique comment procéder en utilisant les formes de communication disponibles pour les systèmes IBM MQ for Windows.

Avant de commencer

Il peut être utile de consulter la rubrique [Exemple de configuration- IBM MQ for Windows](#).

 Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat, Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Pourquoi et quand exécuter cette tâche

Lors de la configuration de la communication pour IBM MQ sur Windows, vous pouvez choisir parmi les types de communication suivants:

- TCP/IP
- LU 6.2
- NetBIOS

Procédure

- Pour plus d'informations sur la configuration de la communication pour votre système Windows, voir la sous-rubrique correspondant au type de communication choisi:
 - [«Définition d'une connexion TCP sous Windows», à la page 280](#)
 - [«Définition d'une connexion LU 6.2 sous Windows», à la page 282](#)
 - [«Définition d'une connexion NetBIOS sous Windows», à la page 284](#)

Toutes les fonctions et fonctions de IBM MQ for Windows ne sont pas disponibles dans les environnements qui utilisent des protocoles de communication autres que TCP/IP. L'élément qui n'est pas disponible est IBM MQ Explorer.

Tâches associées

«Surveillance et contrôle des canaux sur AIX, Linux, and Windows», à la page 268

Pour DQM, vous devez créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éligées. Vous pouvez contrôler les canaux à l'aide de commandes, de programmes, de IBM MQ

Explorer, de fichiers pour les définitions de canal et d'une zone de stockage pour les informations de synchronisation.

«[Configuration des connexions entre le client et le serveur](#)», à la page 17

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

«[Configuration de la communication sur AIX and Linux](#)», à la page 287

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour que cette opération aboutisse, la connexion doit être définie et disponible. Cette section explique comment procéder en utilisant les formes de communication disponibles pour les systèmes IBM MQ for UNIX or Linux .

Référence associée



«[Type de communication à utiliser](#)», à la page 17

Différentes plateformes prennent en charge différents protocoles de communication. Votre choix de protocole de transmission dépend de votre combinaison de IBM MQ MQI client et de plateformes serveur.

Définition d'une connexion TCP sous Windows

Définissez une connexion TCP en configurant un canal à l'extrémité émettrice pour spécifier l'adresse de la cible et en exécutant un programme d'écoute à l'extrémité réceptrice.

Avant de commencer

  Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Fin de l'envoi

Indiquez le nom d'hôte ou l'adresse TCP de la machine cible dans la zone Nom de connexion de la définition de canal.

Le port auquel se connecter est défini par défaut sur 1414. Le numéro de port 1414 est affecté par l'autorité Internet Assigned Numbers à IBM MQ.

Pour utiliser un numéro de port autre que le numéro par défaut, spécifiez-le dans la zone de nom de connexion de la définition d'objet canal:

```
DEFINE CHANNEL('channel name') CHLTYPE(SDR) +
  TRPTYPE(TCP) +
  CONNAME('OS2ROG3(1822)') +
  XMITQ('XMITQ name') +
  REPLACE
```

où OS2ROG3 est le nom DNS du gestionnaire de files d'attente éloignées et 1822 est le port requis. (Il doit s'agir du port sur lequel le programme d'écoute de l'extrémité réceptrice est à l'écoute.)

Un canal en cours d'exécution doit être arrêté et redémarré pour prendre en compte toute modification apportée à la définition d'objet de canal.

Vous pouvez modifier le numéro de port par défaut en le spécifiant dans le fichier .ini pour IBM MQ for Windows:

```
TCP:
Port=1822
```


Remarque : Pour sélectionner le numéro de port TCP/IP à utiliser, IBM MQ utilise le premier numéro de port qu'il trouve dans l'ordre suivant:

1. Numéro de port spécifié explicitement dans la définition de canal ou la ligne de commande. Ce numéro permet de remplacer le numéro de port par défaut pour un canal.
2. Attribut de port spécifié dans la strophe TCP du fichier `.ini`. Ce numéro permet de remplacer le numéro de port par défaut pour un gestionnaire de files d'attente.
3. La valeur par défaut est 1414. Il s'agit du numéro affecté à IBM MQ par l'autorité Internet Assigned Numbers pour les connexions entrantes et sortantes.

Pour plus d'informations sur les valeurs que vous définissez à l'aide de `qm.ini`, voir [Strophes du fichier de configuration pour la mise en file d'attente répartie](#).

Réception sur TCP

Pour démarrer un programme de canal récepteur, un programme d'écoute doit être démarré pour détecter les demandes réseau entrantes et démarrer le canal associé. Vous pouvez utiliser le programme d'écoute IBM MQ.

Les programmes de canal récepteur sont démarrés en réponse à une demande de démarrage du canal émetteur.

Pour démarrer un programme de canal récepteur, un programme d'écoute doit être démarré pour détecter les demandes réseau entrantes et démarrer le canal associé. Vous pouvez utiliser le programme d'écoute IBM MQ.

Pour exécuter le programme d'écoute fourni avec IBM MQ, qui démarre de nouveaux canaux en tant qu'unités d'exécution, utilisez la commande `runmqtsr`.

Exemple de base d'utilisation de la commande `runmqtsr` :

```
runmqtsr -t tcp [-m QMNAME] [-p 1822]
```

Les crochets indiquent des paramètres facultatifs ; QMNAME n'est pas requis pour le gestionnaire de files d'attente par défaut et le numéro de port n'est pas requis si vous utilisez la valeur par défaut (1414). Le numéro de port ne doit pas dépasser 65535.

Remarque : Pour sélectionner le numéro de port TCP/IP à utiliser, IBM MQ utilise le premier numéro de port qu'il trouve dans l'ordre suivant:

1. Numéro de port spécifié explicitement dans la définition de canal ou la ligne de commande. Ce numéro permet de remplacer le numéro de port par défaut pour un canal.
2. Attribut de port spécifié dans la strophe TCP du fichier `.ini`. Ce numéro permet de remplacer le numéro de port par défaut pour un gestionnaire de files d'attente.
3. La valeur par défaut est 1414. Il s'agit du numéro affecté à IBM MQ par l'autorité Internet Assigned Numbers pour les connexions entrantes et sortantes.

Pour des performances optimales, exécutez le programme d'écoute IBM MQ en tant qu'application sécurisée, comme décrit dans «Exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées», à la page 267. Pour plus d'informations sur les applications sécurisées, voir [Restrictions relatives aux applications sécurisées](#).

Utilisation de l'option TCP/IP SO_KEEPALIVE

Si vous souhaitez utiliser l'option Windows `SO_KEEPALIVE`, vous devez ajouter l'entrée suivante à votre registre:

```
TCP:  
KeepAlive=yes
```

Pour plus d'informations sur l'option SO_KEEPALIVE, voir [«Vérification de la disponibilité de l'autre extrémité du canal»](#), à la page 252.

Sous Windows, la valeur de registre HKLM\SYSTEM\CurrentControlSet\Services\Tcpip\Parameters de l'option Windows **KeepAliveTime** contrôle l'intervalle qui s'écoule avant la vérification de la connexion. La valeur par défaut est de deux heures.

Utilisation de l'option en attente du programme d'écoute TCP

Dans TCP, les connexions sont traitées comme incomplètes sauf si un établissement de liaison à trois est effectué entre le serveur et le client. Ces connexions sont appelées demandes de connexion en attente. Une valeur maximale est définie pour ces demandes de connexion en attente et peut être considérée comme un journal des demandes en attente sur le port TCP pour que le programme d'écoute accepte la demande.

Pour plus d'informations et pour connaître la valeur spécifique de Windows, voir [«Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP sous IBM MQ for Multiplatforms»](#), à la page 290 .

Windows Définition d'une connexion LU 6.2 sous Windows

SNA doit être configuré pour qu'une conversation LU 6.2 puisse être établie entre les deux machines.

Une fois le SNA configuré, procédez comme suit.

Pour plus d'informations, voir le tableau suivant.

<i>Tableau 23. Paramètres sur le système Windows local pour une plateforme de gestionnaire de files d'attente éloignées</i>		
Plateforme distante	TPNAME	TPPATH
z/OS ou MVS/ESA sans CICS	Identique à la valeur indiquée dans les informations complémentaires correspondantes sur le gestionnaire de files d'attente éloignées.	-
z/OS ou MVS/ESA à l'aide de CICS	CKRC (émetteur) CKSV (demandeur) CKRC (serveur)	-
IBM i	Identique à la valeur de comparaison dans le poste de routage sur le système IBM i .	-
Systèmes AIX and Linux	Identique à la valeur indiquée dans les informations complémentaires correspondantes sur le gestionnaire de files d'attente éloignées.	<i>MQ_INSTALLATION_PATH</i> /bin/amqcrs6a
Windows	Comme indiqué dans la commande Windows Exécuter le programme d'écoute ou dans le programme de transaction facturable qui a été défini à l'aide de TpSetup sous Windows.	<i>MQ_INSTALLATION_PATH</i> \bin\amqcrs6a

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

Si vous disposez de plusieurs gestionnaires de files d'attente sur la même machine, vérifiez que les noms de programme transactionnel dans les définitions de canal sont uniques.

Pour les informations les plus récentes sur la configuration de AnyNet SNA sur TCP/IP, voir la documentation en ligne IBM suivante: [AnyNet SNA sur TCP/IP](#) et [SNA Node Operations](#).

Concepts associés

«Fin de l'envoi sur l'unité logique 6.2 sous Windows», à la page 283

Créez un objet côté CPI-C (destination symbolique) à partir de l'application d'administration du produit LU 6.2 que vous utilisez. Entrez ce nom dans la zone Nom de connexion de la définition de canal. Créez également un lien de LU 6.2 vers le partenaire.

«Réception sur LU 6.2 sous Windows», à la page 283

Les programmes de canal récepteur sont démarrés en réponse à une demande de démarrage du canal émetteur.

Windows

Fin de l'envoi sur l'unité logique 6.2 sous Windows

Créez un objet côté CPI-C (destination symbolique) à partir de l'application d'administration du produit LU 6.2 que vous utilisez. Entrez ce nom dans la zone Nom de connexion de la définition de canal. Créez également un lien de LU 6.2 vers le partenaire.

Dans l'objet côté CPI-C, entrez le nom de LU partenaire sur la machine réceptrice, le nom de TP et le nom de mode. Exemple :

```
Partner LU Name      OS2R0G2
Partner TP Name     recv
Mode Name           #INTER
```

Windows

Réception sur LU 6.2 sous Windows

Les programmes de canal récepteur sont démarrés en réponse à une demande de démarrage du canal émetteur.

Pour démarrer un programme de canal récepteur, un programme d'écoute doit être démarré pour détecter les demandes réseau entrantes et démarrer le canal associé. Vous démarrez ce programme d'écoute à l'aide de la commande RUNMQLSR, en indiquant le TpName à écouter. Vous pouvez également utiliser TpStart sous SNA Server for Windows.

Utilisation de la commande RUNMQLSR

Exemple de commande permettant de démarrer le programme d'écoute:

```
RUNMQLSR -t LU62 -n RECV -m QMNAME
```

où RECV est le TpName spécifié à l'autre extrémité (envoi) comme "TpName à démarrer côté distant". Le paramètre **-m** utilisé dans la dernière partie de cette commande est facultatif et n'est pas requis pour le gestionnaire de files d'attente par défaut.

Il est possible que plusieurs gestionnaires de files d'attente s'exécutent sur une seule machine. Vous devez affecter un nom TpName différent à chaque gestionnaire de files d'attente, puis démarrer un programme d'écoute pour chacun d'eux. Exemple :

```
RUNMQLSR -t LU62 -m QM1 -n TpName1
RUNMQLSR -t LU62 -m QM2 -n TpName2
```

Pour des performances optimales, exécutez le programme d'écoute IBM MQ en tant qu'application sécurisée, comme décrit dans [Exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées](#). Pour plus d'informations sur les applications sécurisées, voir [Restrictions relatives aux applications sécurisées](#).

Vous pouvez arrêter tous les programmes d'écoute IBM MQ exécutés sur un gestionnaire de files d'attente inactif, à l'aide de la commande suivante:

```
ENDMQLSR -m QMNAME
```

Utilisation de Microsoft SNA Server sous Windows

Vous pouvez utiliser TpSetup (à partir du SDK SNA Server) pour définir un programme transactionnel facturable qui exécute ensuite amqcrs6a.exe, ou vous pouvez définir plusieurs valeurs de registre manuellement. Les paramètres à transmettre à amqcrs6a.exe sont les suivants:

```
-m QM -n TpName
```

où *QM* est le nom du gestionnaire de files d'attente et *TpName* est le nom du programme transactionnel. Pour plus d'informations, voir *Microsoft SNA Server APPC Programmers Guide* ou *Microsoft SNA Server CPI-C Programmers Guide*.

Si vous ne spécifiez pas de nom de gestionnaire de files d'attente, le gestionnaire de files d'attente par défaut est utilisé.

Windows Définition d'une connexion NetBIOS sous Windows

Une connexion NetBIOS s'applique uniquement à un client et un serveur exécutant Windows. IBM MQ utilise trois types de ressource NetBIOS lors de l'établissement d'une connexion NetBIOS à un autre produit IBM MQ : les sessions, les commandes et les noms. Chacune de ces ressources a une limite, qui est établie par défaut ou par choix lors de l'installation de NetBIOS.

Chaque canal en cours d'exécution, quel que soit son type, utilise une session NetBIOS et une commande NetBIOS. L'implémentation IBM NetBIOS permet à plusieurs processus d'utiliser le même nom NetBIOS local. Par conséquent, un seul nom NetBIOS doit être disponible pour être utilisé par IBM MQ. Les implémentations d'autres fournisseurs, par exemple l'émulation NetBIOS de Novell, nécessitent un nom local différent par processus. Vérifiez vos exigences dans la documentation du produit NetBIOS que vous utilisez.

Dans tous les cas, assurez-vous que des ressources suffisantes de chaque type sont déjà disponibles ou augmentez les valeurs maximales spécifiées dans la configuration. Toute modification des valeurs nécessite un redémarrage du système.

Lors du démarrage du système, le pilote de périphérique NetBIOS affiche le nombre de sessions, de commandes et de noms pouvant être utilisés par les applications. Ces ressources sont disponibles pour toute application NetBIOS exécutée sur le même système. Par conséquent, il est possible que d'autres applications consomment ces ressources avant que IBM MQ n'ait besoin de les acquérir. Votre administrateur de réseau local doit être en mesure de clarifier ce point pour vous.

Concepts associés

«Définition du nom NetBIOS IBM MQ local», à la page 284

Le nom NetBIOS local utilisé par les processus de canal IBM MQ peut être spécifié de trois manières.

«Établissement des limites de sessions, de commandes et de noms NetBIOS du gestionnaire de files d'attente», à la page 285

Les limites du gestionnaire de files d'attente pour les sessions, les commandes et les noms NetBIOS peuvent être spécifiées de deux manières.

«Établissement du numéro de carte de réseau local», à la page 286

Pour que les canaux fonctionnent correctement sur NetBIOS, la prise en charge de la carte à chaque extrémité doit être compatible. IBM MQ vous permet de contrôler le choix du numéro d'adaptateur LAN (LANA) en utilisant la valeur AdapterNum dans la strophe NETBIOS de votre fichier qm.ini et en spécifiant le paramètre **-a** dans la commande runmqsr.

«Lancement de la connexion NetBIOS», à la page 286

Définition des étapes nécessaires à l'établissement d'une connexion.

«Définition du programme d'écoute cible pour la connexion NetBIOS», à la page 286

Définition des étapes à effectuer à l'extrémité réceptrice de la connexion NetBIOS.

Windows Définition du nom NetBIOS IBM MQ local

Le nom NetBIOS local utilisé par les processus de canal IBM MQ peut être spécifié de trois manières.

Par ordre de priorité, les trois méthodes sont les suivantes:

1. La valeur spécifiée dans le paramètre **-l** de la commande **runmqlsr**, par exemple:

```
runmqlsr -t netbios -l my_station
```

2. La variable d'environnement **MQNAME** avec une valeur établie par la commande:

```
SET MQNAME= my_station
```

Exemple :

```
SET MQNAME=CLIENT1
```

Vous pouvez définir la valeur **MQNAME** pour chaque processus. Vous pouvez également le définir au niveau du système dans le registre Windows .

Si vous utilisez une implémentation NetBIOS qui requiert des noms uniques, vous devez émettre une commande **SET MQNAME** dans chaque fenêtre dans laquelle un processus IBM MQ est démarré. La valeur **MQNAME** est arbitraire, mais elle doit être unique pour chaque processus.

3. La strophe NETBIOS du fichier de configuration du gestionnaire de files d'attente `qm.ini`. Exemple :

```
NETBIOS:  
LocalName= my_station
```

Remarque :

1. En raison des variations dans l'implémentation des produits NetBIOS pris en charge, il est conseillé de rendre chaque nom NetBIOS unique dans le réseau. Si vous ne le faites pas, des résultats imprévisibles peuvent se produire. Si vous rencontrez des problèmes lors de l'établissement d'un canal NetBIOS et que le journal des erreurs du gestionnaire de files d'attente contient des messages d'erreur indiquant un code retour NetBIOS X'15', passez en revue l'utilisation des noms NetBIOS .
2. Sous Windows, vous ne pouvez pas utiliser votre nom de machine comme nom NetBIOS car Windows l'utilise déjà.
3. L'initiation d'un canal émetteur requiert la spécification d'un nom NetBIOS à l'aide de la variable d'environnement **MQNAME** ou de l'élément **LocalName** dans le fichier `qm.ini` .

Windows *Etablissement des limites de sessions, de commandes et de noms NetBIOS du gestionnaire de files d'attente*

Les limites du gestionnaire de files d'attente pour les sessions, les commandes et les noms NetBIOS peuvent être spécifiées de deux manières.

Par ordre de priorité, ces méthodes sont les suivantes:

1. Les valeurs spécifiées dans la commande **RUNMQLSR** sont les suivantes:

```
-s Sessions  
-e Names  
-o Commands
```

Si l'opérande **-m** n'est pas spécifié dans la commande, les valeurs s'appliquent uniquement au gestionnaire de files d'attente par défaut.

2. Section **NETBIOS** du fichier de configuration du gestionnaire de files d'attente `qm.ini`. Exemple :

```
NETBIOS:  
NumSess= Qmgr_max_sess
```

```
NumCmds= Qmgr_max_cmds
NumNames= Qmgr_max_names
```

Windows *Etablissement du numéro de carte de réseau local*

Pour que les canaux fonctionnent correctement sur NetBIOS, la prise en charge de la carte à chaque extrémité doit être compatible. IBM MQ vous permet de contrôler le choix du numéro d'adaptateur LAN (LANA) en utilisant la valeur AdapterNum dans la strophe NETBIOS de votre fichier qm.ini et en spécifiant le paramètre **-a** dans la commande runmqsr.

Le numéro de carte de réseau local par défaut utilisé par les connexions IBM MQ for NetBIOS est 0. Vérifiez le nombre utilisé sur votre système comme suit:

Sous Windows, il n'est pas possible d'interroger le numéro de carte de réseau local directement via le système d'exploitation. A la place, vous utilisez LANACFG.EXE , disponible dans Microsoft. La sortie de l'outil affiche les numéros d'adaptateur de réseau local virtuel et leurs liaisons effectives. Pour plus d'informations sur les numéros d'adaptateur LAN, voir l' Microsoft article de la base de connaissances 138037 *HOWTO: Use LANA Numbers in a 32-bit Environment*.

Indiquez la valeur correcte dans la section NETBIOS du fichier de configuration du gestionnaire de files d'attente, qm.ini:

```
NETBIOS:
AdapterNum= n
```

où n est le numéro de carte de réseau local correct pour ce système.

Windows *Lancement de la connexion NetBIOS*

Définition des étapes nécessaires à l'établissement d'une connexion.

Pour initier la connexion, procédez comme suit à la fin de l'envoi:

1. Définissez le nom de la station NetBIOS à l'aide de la valeur MQNAME ou LocalName .
2. Vérifiez le numéro de carte de réseau local utilisé sur votre système et indiquez le fichier correct à l'aide de la commande AdapterNum.
3. Dans la zone ConnectionName de la définition de canal, indiquez le nom NetBIOS utilisé par le programme d'écoute cible. Sous Windows, les canaux NetBIOS doivent être exécutés en tant qu'unités d'exécution. Pour ce faire, indiquez MCATYPE (THREAD) dans la définition de canal.

```
DEFINE CHANNEL (chname) CHLTYPE(SDR) +
TRPTYPE(NETBIOS) +
CONNNAME(your_station) +
XMITQ(xmitq) +
MCATYPE(THREAD) +
REPLACE
```

Windows *Définition du programme d'écoute cible pour la connexion NetBIOS*

Définition des étapes à effectuer à l'extrémité réceptrice de la connexion NetBIOS .

A la réception, procédez comme suit:

1. Définissez le nom de la station NetBIOS à l'aide de la valeur MQNAME ou LocalName .
2. Vérifiez le numéro de carte de réseau local utilisé sur votre système et indiquez le fichier correct à l'aide de la commande AdapterNum.
3. Définissez le canal récepteur:

```
DEFINE CHANNEL (chname) CHLTYPE(RCVR) +
TRPTYPE(NETBIOS) +
REPLACE
```

4. Démarrez le programme d'écoute IBM MQ pour établir la station et permettre de la contacter.
Exemple :

```
RUNMQLSR -t NETBIOS -l your_station [-m qmgr]
```

Cette commande établit `your_station` en tant que station NetBIOS en attente de contact. Le nom de la station NetBIOS doit être unique sur l'ensemble de votre réseau NetBIOS .

Pour des performances optimales, exécutez le programme d'écoute IBM MQ en tant qu'application sécurisée, comme décrit dans [«Exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées»](#), à la page 267. Pour plus d'informations sur les applications sécurisées, voir [Restrictions relatives aux applications sécurisées](#) .

Vous pouvez arrêter tous les programmes d'écoute IBM MQ exécutés sur un gestionnaire de files d'attente inactif, à l'aide de la commande suivante:

```
ENDMQLSR [-m QMNAME]
```

Si vous ne spécifiez pas de nom de gestionnaire de files d'attente, le gestionnaire de files d'attente par défaut est utilisé.

Linux

AIX

Configuration de la communication sur AIX and Linux

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour que cette opération aboutisse, la connexion doit être définie et disponible. Cette section explique comment procéder en utilisant les formes de communication disponibles pour les systèmes IBM MQ for UNIX or Linux .

Avant de commencer

Il peut être utile de consulter les sections suivantes:

- [AIX](#) Exemple de configuration- IBM MQ for AIX
- [Linux](#) Exemple de configuration- IBM MQ pour Linux

MQ Adv.

CD

Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Pourquoi et quand exécuter cette tâche

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour réussir, il est nécessaire que la connexion soit définie et disponible. Cette section explique comment procéder.

Lors de la configuration de la communication pour IBM MQ sur AIX and Linux, vous pouvez choisir parmi les types de communication suivants:


- TCP/IP
- LU 6.2

Chaque définition de canal doit en spécifier une uniquement en tant qu'attribut de protocole de transmission (Type de transport). Un ou plusieurs protocoles peuvent être utilisés par un gestionnaire de files d'attente.

Pour IBM MQ MQI clients, il peut être utile de disposer de canaux alternatifs utilisant des protocoles de transmission différents. Voir [IBM MQ MQI clients](#).

Procédure

Pour plus d'informations sur la configuration de la communication pour votre système AIX ou Linux , voir la sous-rubrique correspondant au type de communication choisi:

- [«Définition d'une connexion TCP sous AIX and Linux»](#), à la page 288
- [«Définition d'une connexion LU 6.2 sous AIX and Linux»](#), à la page 292
-  [«Définition d'une connexion Aspera gateway sur les plateformes Linux ou Windows»](#), à la page 920

Tâches associées

[«Surveillance et contrôle des canaux sur AIX, Linux, and Windows»](#), à la page 268

Pour DQM, vous devez créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Vous pouvez contrôler les canaux à l'aide de commandes, de programmes, de IBM MQ Explorer, de fichiers pour les définitions de canal et d'une zone de stockage pour les informations de synchronisation.

[«Configuration des connexions entre le client et le serveur»](#), à la page 17

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

[«Configuration de la communication sur Windows»](#), à la page 279

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour que cette opération aboutisse, la connexion doit être définie et disponible. Cette section explique comment procéder en utilisant les formes de communication disponibles pour les systèmes IBM MQ for Windows .

Référence associée



[«Type de communication à utiliser»](#), à la page 17

Différentes plateformes prennent en charge différents protocoles de communication. Votre choix de protocole de transmission dépend de votre combinaison de IBM MQ MQI client et de plateformes serveur.

Définition d'une connexion TCP sous AIX and Linux

La définition de canal à l'extrémité émettrice indique l'adresse de la cible. Le programme d'écoute ou le démon inet est configuré pour la connexion à l'extrémité réceptrice.

Avant de commencer

  Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Fin de l'envoi

Indiquez le nom d'hôte ou l'adresse TCP de la machine cible dans la zone Nom de connexion de la définition de canal. Le port auquel se connecter est défini par défaut sur 1414. Le numéro de port 1414 est affecté par l'autorité Internet Assigned Numbers à IBM MQ.

Pour utiliser un numéro de port autre que le numéro par défaut, modifiez la zone de nom de connexion comme suit:

```
Connection Name REMHOST(1822)
```

où REMHOST est le nom d'hôte de la machine distante et 1822 est le numéro de port requis. (Il doit s'agir du port sur lequel le programme d'écoute de l'extrémité réceptrice est à l'écoute.)

Vous pouvez également modifier le numéro de port en le spécifiant dans le fichier de configuration du gestionnaire de files d'attente (qm.ini):

```
TCP:
Port=1822
```

Pour plus d'informations sur les valeurs que vous définissez à l'aide de qm.ini, voir [Strophes du fichier de configuration pour la mise en file d'attente répartie](#).

Réception sur TCP

Vous pouvez utiliser le programme d'écoute TCP/IP, qui est le démon inet (inetd), ou le programme d'écoute IBM MQ .

Certaines distributions Linux utilisent le démon inet étendu (xinetd) à la place du démon inet. Pour plus d'informations sur l'utilisation du démon inet étendu sur un système Linux , voir l' [étape 2](#) de la section [Exemple: configuration de la communication multiplateforme IBM MQ sur Linux](#).

Concepts associés

«Utilisation du programme d'écoute TCP/IP sous AIX and Linux», à la page 289

Pour démarrer les canaux sous AIX and Linux, le fichier `/etc/services` et le fichier `inetd.conf` doivent être édités

«Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP sous IBM MQ for Multiplatforms», à la page 290

Dans TCP, les connexions sont traitées comme incomplètes sauf si un établissement de liaison à trois est effectué entre le serveur et le client. Ces connexions sont appelées demandes de connexion en attente. Une valeur maximale est définie pour ces demandes de connexion en attente et peut être considérée comme un journal des demandes en attente sur le port TCP pour que le programme d'écoute accepte la demande.

«Utilisation du programme d'écoute IBM MQ», à la page 291

Pour exécuter le programme d'écoute fourni avec IBM MQ, qui démarre de nouveaux canaux en tant qu'unités d'exécution, utilisez la commande `runmq1sr` .

«Utilisation de l'option TCP/IP SO_KEEPALIVE», à la page 292

Sur certains systèmes AIX and Linux , vous pouvez définir la durée d'attente TCP avant de vérifier que la connexion est toujours disponible, et la fréquence à laquelle elle tente à nouveau la connexion si la première vérification échoue. Il s'agit soit d'un paramètre réglable du noyau, soit d'un paramètre pouvant être entré sur la ligne de commande.

Utilisation du programme d'écoute TCP/IP sous AIX and Linux

Pour démarrer les canaux sous AIX and Linux, le fichier `/etc/services` et le fichier `inetd.conf` doivent être édités

Suivez ces instructions:

1. Éditez le fichier `/etc/services` :

Remarque : Pour éditer le fichier `/etc/services` , vous devez être connecté en tant que superutilisateur ou root. Vous pouvez modifier cette valeur, mais elle doit correspondre au numéro de port spécifié à la fin de l'envoi.

Ajoutez la ligne suivante au fichier :

```
MQSeries 1414/tcp
```

où 1414 est le numéro de port requis par IBM MQ. Le numéro de port ne doit pas dépasser 65535.

2. Ajoutez une ligne dans le fichier `inetd.conf` pour appeler le programme `amqcrsta`, où `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé:

```
MQSeries stream tcp nowait mqm MQ_INSTALLATION_PATH/bin/amqcrsta amqcrsta  
[-m Queue_Man_Name]
```

Les mises à jour sont actives une fois que `inetd` a relu les fichiers de configuration. Pour ce faire, exécutez les commandes suivantes à partir de l'ID utilisateur `root`:

- **AIX** Sous AIX :

```
refresh -s inetd
```

- **Linux** Sur les systèmes Linux :

```
kill -1 process_number
```

Lorsque le programme d'écoute démarré par `inetd` hérite de l'environnement local de `inetd`, il est possible que MQMDE ne soit pas honoré (fusionné) et qu'il soit placé dans la file d'attente en tant que données de message. Pour vous assurer que MQMDE est honoré, vous devez définir l'environnement local correctement. L'environnement local défini par `inetd` peut ne pas correspondre à celui choisi pour les autres environnements locaux utilisés par les processus IBM MQ . Pour définir l'environnement local:

1. Créez un script shell qui définit les variables d'environnement locales `LANG`, `LC_COLLATE`, `LC_CTYPE`, `LC_MONÉTAIRE`, `LC_NUMERIC`, `LC_TIME` et `LC_MESSAGES` sur l'environnement local utilisé pour les autres processus IBM MQ .
2. Dans le même script shell, appelez le programme d'écoute.
3. Modifiez le fichier `inetd.conf` pour appeler votre script shell à la place du programme d'écoute.

Il est possible d'avoir plusieurs gestionnaires de files d'attente sur le serveur. Vous devez ajouter une ligne à chacun des deux fichiers, pour chacun des gestionnaires de files d'attente. Exemple :

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

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

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


Cela permet d'éviter la génération de messages d'erreur en cas de limitation du nombre de demandes de connexion en attente mises en file d'attente sur un port TCP unique. Pour plus d'informations sur le nombre de demandes de connexion en attente, voir [«Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP sous IBM MQ for Multiplatforms»](#), à la page 290.

Multi *Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP sous IBM MQ for Multiplatforms*

Dans TCP, les connexions sont traitées comme incomplètes sauf si un établissement de liaison à trois est effectué entre le serveur et le client. Ces connexions sont appelées demandes de connexion en attente. Une valeur maximale est définie pour ces demandes de connexion en attente et peut être considérée comme un journal des demandes en attente sur le port TCP pour que le programme d'écoute accepte la demande.

Les valeurs de journal des éléments en attente du programme d'écoute par défaut sont indiquées dans le [Tableau 24](#), à la page 291.

Tableau 24. Nombre maximal de demandes de connexion en attente mises en file d'attente sur un port TCP/IP

Plateforme serveur	Nombre maximal de demandes de connexion
 AIX	100
 Linux	100
 IBM i	255
 Windows	100

Si le journal des éléments en attente atteint les valeurs indiquées dans le [Tableau 24](#), à la page 291, la connexion TCP/IP est rejetée et le canal ne peut pas démarrer.

Pour les canaux MCA, le canal passe à l'état RETRY et relance la connexion ultérieurement.

Toutefois, pour éviter cette erreur, vous pouvez ajouter une entrée dans le fichier `qm.ini` :

```
TCP:
ListenerBacklog = n
```

Cette valeur remplace le nombre maximal par défaut de demandes en attente (voir [Tableau 24](#), à la page 291) pour le programme d'écoute TCP/IP.

Remarque : Certains systèmes d'exploitation prennent en charge une valeur supérieure à la valeur par défaut. Si nécessaire, cette valeur peut être utilisée pour éviter d'atteindre la limite de connexion.

Pour exécuter le programme d'écoute avec l'option Backlog activée:

- Utilisez la commande `runmq1sr -b` ou
- Utilisez la commande MQSC **DEFINE LISTENER** avec l'attribut BACKLOG défini sur la valeur requise.

Pour plus d'informations sur la commande `runmq1sr`, voir [runmq1sr](#). Pour plus d'informations sur la commande `DEFINE LISTENER`, voir [DEFINE LISTENER](#).

Concepts associés

«Using the TCP listener backlog option on z/OS», à la page 1032

When receiving on TCP/IP, a maximum number of outstanding connection requests is set. These outstanding requests can be considered a *backlog* of requests waiting on the TCP/IP port for the listener to accept the request.

  *Utilisation du programme d'écoute IBM MQ*

Pour exécuter le programme d'écoute fourni avec IBM MQ, qui démarre de nouveaux canaux en tant qu'unités d'exécution, utilisez la commande `runmq1sr`.

Exemple :

```
runmq1sr -t tcp [-m QMNAME] [-p 1822]
```

Les crochets indiquent des paramètres facultatifs ; QMNAME n'est pas requis pour le gestionnaire de files d'attente par défaut et le numéro de port n'est pas requis si vous utilisez la valeur par défaut (1414). Le numéro de port ne doit pas dépasser 65535.

Pour des performances optimales, exécutez le programme d'écoute IBM MQ en tant qu'application sécurisée, comme décrit dans «Exécution de canaux et de programmes d'écoute en tant qu'applications sécurisées», à la page 267. Pour plus d'informations sur les applications sécurisées, voir [Restrictions relatives aux applications sécurisées](#).

Vous pouvez arrêter tous les programmes d'écoute IBM MQ exécutés sur un gestionnaire de files d'attente inactif, à l'aide de la commande suivante:

```
endmqlsr [-m QMNAME]
```

Si vous ne spécifiez pas de nom de gestionnaire de files d'attente, le gestionnaire de files d'attente par défaut est utilisé.

Linux AIX Utilisation de l'option TCP/IP SO_KEEPLIVE

Sur certains systèmes AIX and Linux , vous pouvez définir la durée d'attente TCP avant de vérifier que la connexion est toujours disponible, et la fréquence à laquelle elle tente à nouveau la connexion si la première vérification échoue. Il s'agit soit d'un paramètre réglable du noyau, soit d'un paramètre pouvant être entré sur la ligne de commande.

Si vous souhaitez utiliser l'option SO_KEEPLIVE (pour plus d'informations, voir «Vérification de la disponibilité de l'autre extrémité du canal», à la page 252) vous devez ajouter l'entrée suivante à votre fichier de configuration de gestionnaire de files d'attente (qm.ini):

```
TCP:
KeepAlive=yes
```

Pour plus d'informations, voir la documentation de votre système AIX ou Linux .

Linux AIX Définition d'une connexion LU 6.2 sous AIX and Linux

SNA doit être configuré pour qu'une conversation LU 6.2 puisse être établie entre les deux machines.

Pour obtenir les informations les plus récentes sur la configuration de SNA sur TCP/IP, voir la documentation en ligne IBM suivante: [Communications Server](#).

SNA doit être configuré pour qu'une conversation LU 6.2 puisse être établie entre les deux systèmes.

Pour plus d'informations, voir le *Guide de configuration APPC multiplateforme* et le tableau ci-après.

<i>Tableau 25. Paramètres sur le système AIX ou Linux local pour une plateforme de gestionnaire de files d'attente éloignées</i>		
Plateforme distante	TPNAME	TPPATH
z/OS sans CICS	Identique au TPName correspondant dans les informations complémentaires sur le gestionnaire de files d'attente éloignées.	-
z/OS via CICS	CKRC (émetteur) CKSV (demandeur) CKRC (serveur)	-
IBM i	Identique à la valeur de comparaison dans le poste de routage sur le système IBM i .	-
Systèmes AIX and Linux	Identique au TPName correspondant dans les informations complémentaires sur le gestionnaire de files d'attente éloignées.	<code>MQ_INSTALLATION_PATH/bin/amqcrs6a</code>
Windows	Comme indiqué dans la commande Windows Exécuter le programme d'écoute ou dans le programme de transaction facturable qui a été défini à l'aide de TpSetup sous Windows.	<code>MQ_INSTALLATION_PATH\bin\amqcrs6a</code>

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

Si vous disposez de plusieurs gestionnaires de files d'attente sur la même machine, vérifiez que les noms de programme transactionnel dans les définitions de canal sont uniques.

Concepts associés

«Fin de l'envoi sur l'unité logique 6.2 sous AIX and Linux», à la page 293

Sur les systèmes AIX and Linux , créez un objet côté CPI-C (destination symbolique) et entrez ce nom dans la zone Nom de connexion de la définition de canal. Créez également un lien de LU 6.2 vers le partenaire.

«Réception sur LU 6.2 sous AIX and Linux», à la page 293

Sur les systèmes AIX and Linux , créez une connexion d'écoute à l'extrémité réceptrice, un profil de connexion logique LU 6.2 et un profil TPN.

Linux **AIX** *Fin de l'envoi sur l'unité logique 6.2 sous AIX and Linux*

Sur les systèmes AIX and Linux , créez un objet côté CPI-C (destination symbolique) et entrez ce nom dans la zone Nom de connexion de la définition de canal. Créez également un lien de LU 6.2 vers le partenaire.

Dans l'objet côté CPI-C, entrez le nom de LU partenaire sur la machine réceptrice, le nom du programme de transaction et le nom du mode. Exemple :

```
Partner LU Name          REMHOST
Remote TP Name          recv
Service Transaction Program no
Mode Name                #INTER
```

SECURITY PROGRAM est utilisé, lorsqu'il est pris en charge par CPI-C, lorsque IBM MQ tente d'établir une session SNA.

Linux **AIX** *Réception sur LU 6.2 sous AIX and Linux*

Sur les systèmes AIX and Linux , créez une connexion d'écoute à l'extrémité réceptrice, un profil de connexion logique LU 6.2 et un profil TPN.

Dans le profil TPN, entrez le chemin d'accès complet au fichier exécutable et le nom du programme de transaction:

```
Full path to TPN executable  MQ_INSTALLATION_PATH/bin/amqcrs6a
Transaction Program name     recv
User ID                       0
```

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

Sur les systèmes où vous pouvez définir l'ID utilisateur, spécifiez un utilisateur membre du groupe mqm.

AIX Sous AIX, définissez les variables d'environnement APPCTPN (nom de transaction) et APPCLLU (nom de LU locale) (vous pouvez utiliser les panneaux de configuration pour le programme de transaction appelé).

Vous devrez peut-être utiliser un gestionnaire de files d'attente autre que le gestionnaire de files d'attente par défaut. Si tel est le cas, définissez un fichier de commandes qui appelle:

```
amqcrs6a -m Queue_Man_Name
```

puis appelez le fichier de commandes.

IBM i Surveillance et contrôle des canaux sur IBM i

Utilisez les commandes et les panneaux DQM pour créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Chaque gestionnaire de files d'attente dispose d'un

programme DQM permettant de contrôler les interconnexions aux gestionnaires de files d'attente éloignées compatibles.

Pourquoi et quand exécuter cette tâche

La liste suivante est une brève description des composants de la fonction de contrôle de canal:

- Les définitions de canal sont conservées en tant qu'objets de gestionnaire de files d'attente.
- Les commandes de canal constituent un sous-ensemble de l'ensemble de commandes IBM MQ for IBM i.

Utilisez la commande GO CMDMQM pour afficher l'ensemble complet des commandes IBM MQ for IBM i.

- Vous utilisez des panneaux de définition de canal ou des commandes pour:
 - Créer, copier, afficher, modifier et supprimer des définitions de canal
 - Démarrer et arrêter les canaux, exécuter une commande ping, réinitialiser les numéros de séquence des canaux et résoudre les messages en attente de validation lorsque les liens ne peuvent pas être rétablis
 - Afficher les informations de statut sur les canaux
- Les canaux peuvent également être gérés à l'aide de MQSC
- Les canaux peuvent également être gérés à l'aide de IBM MQ Explorer
- Les numéros de séquence et les identificateurs d' *unité de travail logique (LUW)* sont stockés dans le fichier de synchronisation et sont utilisés à des fins de synchronisation de canal.

Vous pouvez utiliser les commandes et les panneaux pour: définir les canaux de transmission de messages et les objets associés, et surveiller et contrôler les canaux de transmission de messages. A l'aide de la clé F4=Prompt , vous pouvez spécifier le gestionnaire de files d'attente approprié. Si vous n'utilisez pas l'invite, le gestionnaire de files d'attente par défaut est utilisé. Avec F4=Prompt, un panneau supplémentaire s'affiche, dans lequel vous pouvez entrer le nom du gestionnaire de files d'attente approprié et parfois d'autres données.

Les objets que vous devez définir avec les panneaux sont les suivants:

- Files d'attente de transmission
- Définition de files d'attente éloignées
- Définition des alias de gestionnaire de files d'attente
- Définition des alias de files d'attente de réponse
- Files d'attente locales de réponse
- Définitions de canal de transmission de messages

Pour plus d'informations sur les concepts impliqués dans l'utilisation de ces objets, voir [«Configuration de la file d'attente répartie»](#), à la page 212.

Les canaux doivent être complètement définis et leurs objets associés doivent exister et être disponibles pour être utilisés avant qu'un canal puisse être démarré.

En outre, la liaison de communication particulière pour chaque canal doit être définie et disponible pour qu'un canal puisse être exécuté. Pour une description de la façon dont les liaisons LU 6.2 et TCP/IP sont définies, consultez le guide de communication spécifique à votre installation.

Procédure

- Pour plus d'informations sur la création et l'utilisation d'objets, voir:
 - [«Création d'objets sous IBM i»](#), à la page 295
 - [«Création d'un canal sous IBM i»](#), à la page 295
 - [«Démarrage d'un canal sur IBM i»](#), à la page 297

- «Sélection d'un canal sur IBM i», à la page 298
- «Exploration d'un canal sur IBM i», à la page 298
- «Attribution d'un nouveau nom à un canal sur IBM i», à la page 300
- «Gestion de l'état des canaux sous IBM i», à la page 300
- «Gestion des choix de canal sur IBM i», à la page 301

Concepts associés

«Configuration de la communication pour IBM i», à la page 307

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour qu'elle aboutisse, il est nécessaire que la connexion soit définie et disponible.

Tâches associées

«Configuration des connexions entre le client et le serveur», à la page 17

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

Référence associée

Exemple de configuration- IBM MQ for IBM i

Exemple de planification de canal de transmission de messages pour IBM MQ for IBM i

Commandes CL IBM MQ for IBM i

IBM i Création d'objets sous IBM i

Vous pouvez utiliser la commande CRTMQMQ pour créer la file d'attente et les objets alias.

Vous pouvez créer les objets file d'attente et alias, tels que les files d'attente de transmission, les définitions de file d'attente éloignée, les définitions d'alias de gestionnaire de files d'attente, les définitions d'alias de file d'attente de réponse et les files d'attente locales de réponse.

Pour obtenir la liste des objets par défaut, voir [Système et objets par défaut](#).

IBM i Création d'un canal sous IBM i

Vous pouvez créer un canal à partir du panneau Créer un canal ou à l'aide de la commande CRTMQMCHL sur la ligne de commande.

Pour créer un canal:

1. Utilisez F6 dans le panneau Work with MQM Channels (WRKMQMCHL).

Vous pouvez également utiliser la commande CRTMQMCHL à partir de la ligne de commande.

Dans les deux cas, le panneau Créer un canal s'affiche. Type :

- Nom du canal dans la zone fournie
- Type de canal de cette extrémité de la liaison

2. Appuyez sur la touche Entrée.

Remarque : Vous devez nommer tous les canaux de votre réseau de manière unique. Comme indiqué dans le [Diagramme de réseau illustrant tous les canaux](#), y compris les noms de gestionnaire de files d'attente source et cible dans le nom de canal est un bon moyen de le faire.

Vos entrées sont validées et les erreurs sont signalées immédiatement. Corrigez les erreurs éventuelles et continuez.

Le panneau des paramètres de canal correspondant au type de canal que vous avez choisi s'affiche. Renseignez les zones avec les informations que vous avez collectées précédemment. Appuyez sur Entrée pour créer le canal.

Vous pouvez choisir le contenu des différentes zones dans les descriptions des panneaux de définition de canal dans les panneaux d'aide et dans Attributs de canal.

```

Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

Channel name . . . . . > CHANNAME_____
Channel type . . . . . > *SDR__      *RCVR, *SDR, *SVR, *RQSTR...
Message Queue Manager name      *DFT_____

Replac_____
e . . . . . *NO_      *NO, *YES
Transport type . . . . . *TCP_____ *LU62, *TCP, *SYSDFTCHL
Text 'description' . . . . . > 'Example Channel Definition'_____

Connection name . . . . . *SYSDFTCHL_____
_____
_____
_____
_____

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

```

Figure 25. Créer un canal (1)

```

Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

Transmission queue . . . . . 'TRANSMISSION_QUEUE_NAME'_____

Message channel agent . . . . . *NONE_____ Name, *SYSDFTCHL, *NONE
Library . . . . . _____ Name
Message channel agent user ID . . . *SYSDFTCHL__ Character value...
Coded Character Set Identifier *SYSDFTCHL__ 0-9999, *SYSDFTCHL
Batch size . . . . . 50_____ 1-9999, *SYSDFTCHL
Disconnect interval . . . . . 6000_____ 1-999999, *SYSDFTCHL
Short retry interval . . . . . 60_____ 0-999999999, *SYSDFTCHL
Short retry count . . . . . 10_____ 0-999999999, *SYSDFTCHL
Long retry interval . . . . . 1200_____ 0-999999999, *SYSDFTCHL
Long retry count . . . . . 999999999__ 0-999999999, *SYSDFTCHL
Security exit . . . . . *NONE_____ Name, *SYSDFTCHL, *NONE
Library . . . . . _____ Name
Security exit user data . . . . . *SYSDFTCHL_____

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

```

Figure 26. Créer un canal (2)

Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

```
Send exit . . . . . *NONE_____ Name, *SYSDFTCHL, *NONE
Library . . . . . _____ Name
+ for more values
Send exit user data . . . . . _____
+ for more values
Receive exit . . . . . *NONE_____ Name, *SYSDFTCHL, *NONE
Library . . . . . _____ Name
+ for more values
-----
Receive exit user data . . . . . _____
+ for more values
Message exit . . . . . *NONE_____ Name, *SYSDFTCHL, *NONE
Library . . . . . _____ Name
+ for more values
-----
More...
F3=Exit F4=Prompt F5=Refresh F12=Cancel F13=How to use this display
F24=More keys
```

Figure 27. Créer un canal (3)

Create MQM Channel (CRTMQMCHL)

Type choices, press Enter.

```
Message exit user data . . . . . _____
+ for more values
Convert message . . . . . *SYSDFTCHL_ *YES, *NO, *SYSDFTCHL
Sequence number wrap . . . . . 99999999__ 100-99999999, *SYSDFTCHL
Maximum message length . . . . . 4194304___ 0-4194304, *SYSDFTCHL
Heartbeat interval . . . . . 300_____ 0-999999999, *SYSDFTCHL
Non Persistent Message Speed . . *FAST_____ *FAST, *NORMAL, *SYSDFTCHL
Password . . . . . *SYSDFTCHL_ Character value, *BLANK...
Task User Profile . . . . . *SYSDFTCHL_ Character value, *BLANK...
Transaction Program Name . . . . . *SYSDFTCHL
```

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

Figure 28. Créer un canal (4)

Démarrage d'un canal sur IBM i

Vous pouvez démarrer un canal à partir du panneau Work with Channels ou à l'aide de la commande STRMQMCHL sur la ligne de commande.

Les programmes d'écoute sont valides uniquement pour TCP. Pour les programmes d'écoute SNA, vous devez configurer votre sous-système de communications.

Pour que les applications puissent échanger des messages, vous devez démarrer un programme d'écoute pour les connexions entrantes à l'aide de la commande STRMQMLSR.

Pour les connexions sortantes, vous devez démarrer le canal de l'une des manières suivantes:

1. Utilisez la commande CL STRMQMCHL, en spécifiant le nom de canal, pour démarrer le canal en tant que processus ou unité d'exécution, en fonction du paramètre MCATYPE. (Si les canaux sont démarrés en tant qu'unités d'exécution, il s'agit des unités d'exécution d'un initiateur de canal.)

```
STRMQMCHL CHLNAME(QM1.TO.QM2) MQNAME(MYQMGR)
```

2. Utilisez un initiateur de canal pour déclencher le canal. Un initiateur de canal est démarré automatiquement lorsque le gestionnaire de files d'attente est démarré. Ce démarrage automatique peut être éliminé en modifiant la strophe chinit dans le fichier qm.ini de ce gestionnaire de files d'attente.
3. Utilisez la commande WRKMQMCHL pour lancer le panneau Gestion des canaux et choisissez l'option 14 pour démarrer un canal.

IBM i Sélection d'un canal sur IBM i

Vous pouvez sélectionner un canal dans le panneau Gestion des canaux.

Pour sélectionner un canal, utilisez la commande WRKMQMCHL pour commencer à l'écran Gestion des canaux:

1. Placez le curseur sur la zone d'option associée au nom de canal requis.
2. Entrez un numéro d'option.
3. Appuyez sur Entrée pour activer votre choix.

Si vous sélectionnez plusieurs canaux, les options sont activées dans l'ordre.

```
Work with MQM Channels
Queue Manager Name . . : CNX

Type options, press Enter.
2=Change 3=Copy 4=Delete 5=Display 8=Work with Status 13=Ping
14=Start 15=End 16=Reset 17=Resolve

Opt Name          Type      Transport  Status
CHLNIC            *RCVR    *TCP       INACTIVE
CORSAIR.TO.MUSTANG *SDR     *LU62      INACTIVE
FV.CHANNEL.MC.DJE1 *RCVR    *TCP       INACTIVE
FV.CHANNEL.MC.DJE2 *SDR     *TCP       INACTIVE
FV.CHANNEL.MC.DJE3 *RQSTR   *TCP       INACTIVE
FV.CHANNEL.MC.DJE4 *SVR     *TCP       INACTIVE
FV.CHANNEL.PETER  *RCVR    *TCP       INACTIVE
FV.CHANNEL.PETER.LU *RCVR    *LU62      INACTIVE
FV.CHANNEL.PETER.LU1 *RCVR    *LU62      INACTIVE
More...
Parameters or command
===>
F3=Exit F4=Prompt F5=Refresh F6=Create F9=Retrieve F12=Cancel
F21=Print
```

Figure 29. Gestion des canaux

IBM i Exploration d'un canal sur IBM i

Vous pouvez parcourir un canal à partir du panneau Display Channel ou à l'aide de la commande DSPMQMCHL sur la ligne de commande.

Pour parcourir les paramètres d'un canal, utilisez la commande WRKMQMCHL pour commencer dans le panneau Display Channel:

1. Tapez l'option 5 (Afficher) sur le nom de canal requis.
2. Appuyez sur Entrée pour activer votre choix.

Si vous sélectionnez plusieurs canaux, ils sont présentés dans l'ordre.

Vous pouvez également utiliser la commande DSPMQMCHL à partir de la ligne de commande.

Le panneau de canal d'affichage approprié s'affiche alors avec les détails des paramètres en cours pour le canal. Les zones sont décrites dans [Attributs de canal](#).

```
Display MQM Channel

Channel name . . . . . : ST.JST.2T01
Queue Manager Name . . . . . : QMREL
Channel type . . . . . : *SDR
Transport type . . . . . : *TCP
Text 'description' . . . . . : John's sender to WINSDOA1

Connection name . . . . . : MUSTANG

Transmission queue . . . . . : WINSDOA1

Message channel agent . . . . . :
Library . . . . . :
Message channel agent user ID : *NONE
Batch interval . . . . . : 0
Batch size . . . . . : 50
Disconnect interval . . . . . : 6000

F3=Exit F12=Cancel F21=Print
```

Figure 30. Affichage d'un canal TCP/IP (1)

```
Display MQM Channel

Short retry interval . . . . . : 60
Short retry count . . . . . : 10
Long retry interval . . . . . : 6000
Long retry count . . . . . : 10
Security exit . . . . . :
Library . . . . . :
Security exit user data . . . . . :
Send exit . . . . . :
Library . . . . . :
Send exit user data . . . . . :
Receive exit . . . . . :
Library . . . . . :
Receive exit user data . . . . . :
Message exit . . . . . :
Library . . . . . :
Message exit user data . . . . . :
More...

F3=Exit F12=Cancel F21=Print
```

Figure 31. Affichage d'un canal TCP/IP (2)

```
Display MQM Channel
Sequence number wrap . . . . . : 999999999
Maximum message length . . . . : 10000
Convert message . . . . . : *NO
Heartbeat interval . . . . . : 300
Nonpersistent message speed . . *FAST
```

Bottom

F3=Exit F12=Cancel F21=Print

Figure 32. Affichage d'un canal TCP/IP (3)

Attribution d'un nouveau nom à un canal sur IBM i

Vous pouvez renommer un canal à partir du panneau Gestion des canaux.

Pour renommer un canal de transmission de messages, commencez par le panneau Gestion des canaux:

1. Arrêtez le canal.
2. Utilisez l'option 3 (Copier) pour créer un doublon avec le nouveau nom.
3. Utilisez l'option 5 (Afficher) pour vérifier qu'elle a été créée correctement.
4. Utilisez l'option 4 (Supprimer) pour supprimer le canal d'origine.

Si vous décidez de renommer un canal de transmission de messages, assurez-vous que les deux extrémités de canal sont renommées en même temps.

Gestion de l'état des canaux sous IBM i

Vous pouvez utiliser le statut du canal à partir du panneau Work with Channel Status.

Utilisez la commande WRKMQMTCSPS pour afficher le premier d'un ensemble de panneaux indiquant l'état de vos canaux. Vous pouvez afficher les panneaux d'état dans l'ordre lorsque vous sélectionnez Change-view (F11).

Sinon, la sélection de l'option 8 (Work with Status) dans le panneau Work with MQM Channels affiche également le premier panneau d'état.

MQSeries Work with Channel Status

Type options, press Enter.

5=Display 13=Ping 14=Start 15=End 16=Reset 17=Resolve

Opt Name	Connection	Indoubt	Last Seq
CARTS_CORSAIR_CHAN	GBIBMIYA.WINSDOA1	NO	1
CHLNIC	9.20.2.213	NO	3
FV.CHANNEL.PETER2	9.20.2.213	NO	6225
JST.1.2	9.20.2.201	NO	28
MP_MUST_TO_CORS	9.20.2.213	NO	100
MUSTANG.TO.CORSAIR	GBIBMIYA.WINSDOA1	NO	10
MP_CORS_TO_MUST	9.20.2.213	NO	101
JST.2.3	9.5.7.126	NO	32
PF_WINSDOA1_LU62	GBIBMIYA.IYA80020	NO	54
PF_WINSDOA1_LU62	GBIBMIYA.WINSDOA1	NO	500
ST.JCW.EXIT.2T01.CHL	9.20.2.213	NO	216

Bottom

Parameters or command

==>

F3=Exit F4=Prompt F5=Refresh F6=Create F9=Retrieve F11=Change view

F12=Cancel F21=Print

Figure 33. Premier de l'ensemble de panneaux d'état de canal

Les options disponibles dans le panneau Work with Channel Status sont les suivantes:

Option de menu	Description
5=Display	Affiche les paramètres du canal.
13=Ping	Lance une action Ping, le cas échéant.
14=Start	Démarre le canal.
15=End	Arrête le canal.
16=Reset	Réinitialise le numéro de séquence du canal.
17=Resolve	Résout manuellement une situation de canal en attente de validation.

IBM i Gestion des choix de canal sur IBM i

Le panneau Gestion des canaux est accessible à l'aide de la commande WRKMQMCHL. Il vous permet de surveiller l'état de tous les canaux répertoriés et d'exécuter des commandes sur les canaux sélectionnés.

Les options disponibles dans le panneau Work with Channel sont les suivantes:

Option de menu	Description
<u>«2=Change», à la page 302</u>	Modifie les attributs d'un canal.
<u>«3=Copy», à la page 302</u>	Copie les attributs d'un canal vers un nouveau canal.
<u>«4=Delete», à la page 302</u>	Supprime un canal.
<u>«5=Display», à la page 302</u>	Affiche les paramètres en cours du canal.
<u>«6=Create», à la page 302</u>	Affiche le panneau Créer un canal
<u>«8=Work avec le statut», à la page 303</u>	Affiche les panneaux d'état du canal.

Option de menu	Description
<u>«13=Ping», à la page 304</u>	Exécute la fonction Ping pour tester la connexion au système adjacent en échangeant un message de données fixe avec l'extrémité distante.
<u>«14=Start», à la page 304</u>	Démarre le canal sélectionné ou réinitialise un canal récepteur désactivé.
<u>«15=End», à la page 305</u>	Demande la fermeture du canal.
<u>«16=Reset», à la page 306</u>	Demande au canal de réinitialiser les numéros de séquence à cette extrémité de la liaison. Les nombres doivent être égaux aux deux extrémités pour que le canal démarre.
<u>«17=Resolve», à la page 306</u>	Demande au canal de résoudre les messages en attente de validation sans établir de connexion à l'autre extrémité.
<u>«18=Afficher droits d'accès», à la page 306</u>	Affiche les droits sur les objets IBM MQ
<u>«19=Octroyer droits d'accès», à la page 307</u>	Accorde des droits sur les objets IBM MQ
<u>«20=Révoquer droits d'accès», à la page 307</u>	Révoque les droits sur les objets IBM MQ
<u>«21=Retrouver l'objet», à la page 307</u>	Récupère l'objet IBM MQ
<u>«22 = Enregistrement d'image», à la page 307</u>	Enregistre l'image d'objet IBM MQ

IBM i 2=Change

Utilisez l'option Modifier pour modifier une définition de canal existante.

L'option Modifier ou la commande CHGMQMCHL modifie une définition de canal existante, à l'exception du nom de canal. Tapez sur les zones à modifier dans le panneau de définition de canal, puis sauvegardez la définition mise à jour en appuyant sur Entrée.

IBM i 3=Copy

Utilisez l'option Copier pour copier un canal existant.

L'option Copier utilise la commande CPYMQMCHL pour copier un canal existant. Le panneau Copier vous permet de définir le nouveau nom de canal. Toutefois, vous devez limiter les caractères utilisés aux caractères valides pour les noms d'objet IBM i ; voir [Administration IBM MQ for IBM i](#).

Appuyez sur Entrée dans le panneau Copier pour afficher les détails des paramètres en cours. Vous pouvez modifier les nouveaux paramètres de canal. Sauvegardez la nouvelle définition de canal en appuyant sur Entrée.

IBM i 4>Delete

Utilisez l'option Supprimer pour supprimer le canal sélectionné.

Un panneau s'affiche pour vous permettre de confirmer ou d'annuler votre demande.

IBM i 5=Display

Utilisez l'option Afficher pour afficher les définitions en cours du canal.

Cette option affiche le panneau avec les zones affichant les valeurs en cours des paramètres et protégées contre les entrées utilisateur.

IBM i 6=Create

Utilisez l'option Créer pour afficher le panneau Créer un canal.

Utilisez l'option Créer ou entrez la commande CRTMQMCHL à partir de la ligne de commande pour obtenir le panneau Créer un canal. Il existe des exemples de panneaux Créer un canal, à partir de [Figure 25](#), à la [page 296](#).

Ce panneau permet de créer une définition de canal à partir d'un écran de zones remplies de valeurs par défaut fournies par IBM MQ for IBM i. Entrez le nom du canal, sélectionnez le type de canal que vous créez et la méthode de communication à utiliser.

Lorsque vous appuyez sur Entrée, le panneau s'affiche. Entrez les informations dans toutes les zones obligatoires de ce panneau et dans les panneaux restants, puis sauvegardez la définition en appuyant sur la touche Entrée.

Le nom de canal doit être identique aux deux extrémités du canal et unique au sein du réseau. Toutefois, vous devez limiter les caractères utilisés à ceux qui sont valides pour les noms d'objet IBM MQ for IBM i .

Tous les panneaux ont des valeurs par défaut fournies par IBM MQ for IBM i pour certaines zones. Vous pouvez personnaliser ces valeurs ou les modifier lorsque vous créez ou copiez des canaux. Pour personnaliser les valeurs, voir *IBM MQ for IBM i System Administration*.

Vous pouvez créer votre propre ensemble de valeurs par défaut de canal en configurant des canaux factices avec les valeurs par défaut requises pour chaque type de canal et en les copiant chaque fois que vous souhaitez créer de nouvelles définitions de canal.

Référence associée

[Attributs de canal](#)

8=Work avec le statut

Utilisez l'option Gestion de l'état pour afficher des informations détaillées sur l'état des canaux.

La colonne de statut vous indique si le canal est actif ou inactif et s'affiche en continu dans le panneau Work with MQM Channels. Utilisez l'option 8 (Gestion de l'état) pour afficher plus d'informations sur l'état. Vous pouvez également afficher ces informations à partir de la ligne de commande à l'aide de la commande WRKMQMTCSPS. Voir [«Gestion de l'état des canaux sous IBM i»](#), à la [page 300](#).

- Nom du canal
- Type de canal
- Statut de canal
- Instance de canal
- Gestionnaire de files d'attente éloignées
- Nom de la file d'attente de transmission
- Nom de la connexion de communication
- Statut en attente de validation du canal
- Dernier numéro de séquence
- Nombre de messages en attente de validation
- Numéro de séquence en attente de validation
- Nombre de messages dans la file d'attente de transmission
- Identificateur d'unité d'oeuvre logique
- Identificateur d'unité d'oeuvre logique en attente de validation
- Sous-état du canal
- Contrôle des canaux
- Compression d'en-tête
- Compression de message
- Indicateur de temps de compression
- Indicateur de taux de compression

- Indicateur de temps de file d'attente de transmission
- Indicateur de temps réseau
- Indicateur de temps de sortie
- Indicateur de taille de lot
- Conversations partagées en cours
- Nombre maximal de conversations partagées

IBM i **13=Ping**

Utilisez l'option Ping pour échanger un message de données fixe avec l'extrémité distante.

Une commande PING IBM MQ réussie donne au superviseur système la certitude que le canal est disponible et qu'il fonctionne.

La commande ping n'implique pas l'utilisation de files d'attente de transmission et de files d'attente cible. Il utilise des définitions de canal, la liaison de communication associée et la configuration du réseau.

Il est disponible uniquement à partir des canaux émetteur et serveur. Le canal correspondant est démarré du côté opposé de la liaison, et effectue la négociation des paramètres de démarrage. Les erreurs sont notifiées normalement.

Le résultat de l'échange de messages est présenté dans le panneau Ping pour vous. Il s'agit du texte du message renvoyé, ainsi que de l'heure à laquelle le message a été envoyé et de l'heure à laquelle la réponse a été reçue.

Commande ping avec LU 6.2

Lorsque la commande Ping est appelée dans IBM MQ for IBM i, elle est exécutée avec l'ID utilisateur de l'utilisateur demandant la fonction, alors que la méthode d'exécution normale d'un programme de canal consiste à utiliser l'ID utilisateur QMQM pour les programmes de canal. L'ID utilisateur est transmis au côté récepteur et doit être valide sur le côté récepteur pour que la conversation LU 6.2 puisse être allouée.

IBM i **14=Start**

Utilisez l'option Démarrer pour démarrer un canal manuellement.

L'option Démarrer est disponible pour les canaux émetteur, serveur et demandeur. Il n'est pas nécessaire lorsqu'un canal a été configuré avec le déclenchement du gestionnaire de files d'attente.

L'option de démarrage est également utilisée pour les canaux récepteur, serveur-connexion, émetteur de cluster et récepteur de cluster. Le démarrage d'un canal récepteur à l'état STOPPED signifie qu'il peut être démarré à partir du canal distant.

Une fois démarré, l'agent MCA émetteur lit le fichier de définition de canal et ouvre la file d'attente de transmission. Une séquence de démarrage de canal est émise, qui démarre à distance l'agent MCA correspondant du canal récepteur ou serveur. Une fois qu'ils ont été démarrés, les processus de l'expéditeur et du serveur attendent les messages arrivant dans la file d'attente de transmission et les transmettent à mesure qu'ils arrivent.

Lorsque vous utilisez le déclenchement, vous devez démarrer le processus de déclenchement en continu pour surveiller la file d'attente d'initialisation. La commande STRMQMCHLI peut être utilisée pour démarrer le processus.

A l'extrémité éloignée d'un canal, le processus de réception peut être lancé en réponse à un démarrage de canal à partir de l'extrémité émettrice. Cette méthode est différente pour les canaux LU 6.2 et TCP/IP connectés:

- Les canaux connectés à l'unité logique 6.2 ne nécessitent aucune action explicite à l'extrémité réceptrice d'un canal.

- Les canaux connectés TCP nécessitent qu'un processus de programme d'écoute s'exécute en continu. Ce processus attend les demandes de démarrage de canal de l'extrémité distante de la liaison et démarre le processus défini dans les définitions de canal pour cette connexion.

Lorsque le système distant est IBM i, vous pouvez utiliser la commande STRMQMLSR.

L'utilisation de l'option Démarrer entraîne toujours la resynchronisation du canal, si nécessaire.

Pour que le démarrage aboutisse:

- Des définitions de canal, locales et distantes doivent exister. S'il n'existe pas de définition de canal appropriée pour un canal récepteur ou de connexion serveur, un canal par défaut est créé automatiquement si le canal est défini automatiquement. Voir [Programme d'exit de définition automatique de canal](#).
- La file d'attente de transmission doit exister, être activée pour les opérations GET et ne pas être utilisée par d'autres canaux.
- Les agents MCA, locaux et distants, doivent exister.
- La liaison de communication doit être disponible.
- Les gestionnaires de files d'attente doivent être en cours d'exécution, locaux et distants.
- Le canal de transmission de messages doit être inactif.

Pour transférer des messages, des files d'attente éloignées et des définitions de files d'attente éloignées doivent exister.

Un message est renvoyé au panneau confirmant que la demande de démarrage d'un canal a été acceptée. Pour confirmer que le processus de démarrage a abouti, consultez le journal système ou appuyez sur F5 (régénérez l'écran).

▶ IBM i **15=End**

Utiliser Fin pour arrêter l'activité du canal

Utilisez l'option Fin pour demander au canal d'arrêter l'activité. Le canal n'envoie plus de messages.

Sélectionnez F4 avant d'appuyer sur la touche Entrée pour indiquer si le canal devient STOPPED ou INACTIVE et s'il doit être arrêté à l'aide d'une commande CONTROLLED ou d'une commande IMMEDIATE. Un canal arrêté doit être redémarré par l'opérateur pour redevenir actif. Un canal inactif peut être déclenché.

stop immediate

Utilisez l'option Arrêter immédiatement pour arrêter un canal sans terminer une unité de travail.

Cette option met fin au processus de canal. Par conséquent, le canal ne termine pas le traitement du lot de messages en cours et ne peut donc pas laisser le canal en attente de validation. En général, il est préférable pour les opérateurs d'utiliser l'option d'arrêt contrôlé.

Arrêt contrôlé

Utilisez l'option Arrêt contrôlé pour arrêter un canal à la fin de l'unité de travail en cours.

Ce choix demande au canal de se fermer de manière ordonnée ; le lot de messages en cours est terminé, et la procédure de point de synchronisation est effectuée avec l'autre extrémité du canal.

Redémarrage des canaux arrêtés

Lorsqu'un canal passe à l'état STOPPED, vous devez le redémarrer manuellement. Vous pouvez redémarrer le canal de l'une des manières suivantes:

- A l'aide de la commande **START CHANNEL MQSC**.
- A l'aide de la commande PCF **Start Channel** .

- A l'aide de la IBM MQ Explorer.
- **z/OS** Sous z/OS, à l'aide du panneau Démarrer un canal.
- **IBM i** Sous IBM i, à l'aide de la commande **STRMQMCHL CL** ou de l'option **START** du panneau WRKMQMCHL.

Pour les canaux émetteurs ou serveurs, lorsque le canal est passé à l'état STOPPED, la file d'attente de transmission associée a été définie sur GET (DISABLED) et le déclenchement a été désactivé. Lorsque la demande de démarrage est reçue, ces attributs sont réinitialisés automatiquement.

z/OS Si l'initiateur de canal s'arrête alors qu'un canal est à l'état REESSAYEZ ou STOPPED, l'état du canal est mémorisé lors du redémarrage de l'initiateur de canal. Toutefois, le statut du canal SVRCONN est réinitialisé si l'initiateur de canal s'arrête alors que le canal est à l'état STOPPED.

Multi Si le gestionnaire de files d'attente s'arrête alors qu'un canal est à l'état REESSAYEZ ou STOPPED, l'état du canal est mémorisé lors du redémarrage du gestionnaire de files d'attente. A partir de la IBM MQ 8.0, cela s'applique également aux canaux SVRCONN. Auparavant, le statut de canal pour le type de canal SVRCONN était réinitialisé si l'initiateur de canal était arrêté alors que le canal était à l'état STOPPED.

IBM i **16=Reset**

Utilisez l'option de réinitialisation pour forcer une nouvelle séquence de messages.

L'option de réinitialisation modifie le numéro de séquence du message. Utilisez-la avec précaution et uniquement après avoir utilisé l'option Résoudre pour résoudre les situations en attente de validation. Cette option est disponible uniquement sur le canal émetteur ou serveur. Le premier message démarre la nouvelle séquence lors du prochain démarrage du canal.

IBM i **17=Resolve**

Utilisez l'option Resolve pour forcer une validation locale ou l'annulation de messages en attente de validation dans une file d'attente de transmission.

Utilisez l'option Résoudre lorsque des messages sont conservés en attente de validation par un expéditeur ou un serveur, par exemple parce qu'une extrémité du lien s'est arrêtée et qu'il n'y a aucune perspective de récupération. L'option de résolution accepte l'un des deux paramètres suivants: BACKOUT ou COMMIT. L'annulation restaure les messages dans la file d'attente de transmission, tandis que la validation les supprime.

Le programme de canal ne tente pas d'établir une session avec un partenaire. A la place, il détermine l'identificateur d'unité d'oeuvre logique (LUWID) qui représente les messages en attente de validation. Il émet ensuite, comme demandé, soit:

- BACKOUT pour restaurer les messages dans la file d'attente de transmission ; ou
- COMMIT pour supprimer les messages de la file d'attente de transmission.

Pour que la résolution aboutisse:

- Le canal doit être inactif
- Le canal doit être en attente de validation
- Le type de canal doit être émetteur ou serveur
- La définition de canal, locale, doit exister
- Le gestionnaire de files d'attente doit être en cours d'exécution, local

IBM i **18=Afficher droits d'accès**

L'option Afficher les droits permet d'afficher les actions qu'un utilisateur est autorisé à effectuer sur un objet IBM MQ spécifique.

Pour un objet choisi et un utilisateur, la commande DSPMQAUT affiche les autorisations dont dispose l'utilisateur pour effectuer des actions sur un objet IBM MQ . Si l'utilisateur est membre de plusieurs groupes, la commande affiche l'autorisation combinée de tous les groupes sur l'objet.

IBM i 19=Octroyer droits d'accès

Utilisez l'option Accorder les droits pour accorder les droits permettant d'effectuer des actions sur les objets IBM MQ à un autre utilisateur ou groupe d'utilisateurs.

La commande GRMQMAUT est disponible uniquement pour les utilisateurs du groupe QMQMADM. Un utilisateur de QMQMADM accorde des droits à d'autres utilisateurs pour effectuer des actions sur les objets IBM MQ nommés dans la commande, soit en identifiant les utilisateurs par leur nom, soit en accordant des droits à tous les utilisateurs de *PUBLIC.

IBM i 20=Révoquer droits d'accès

Utilisez le droit de révocation pour supprimer l'autorisation d'effectuer des actions sur les objets des utilisateurs.

La commande RVKMQMAUT est disponible uniquement pour les utilisateurs du groupe QMQMADM. Un utilisateur du groupe QMQMADM retire aux autres utilisateurs le droit d'effectuer des actions sur les objets IBM MQ nommés dans la commande en identifiant les utilisateurs par leur nom ou en révoquant les droits de tous les utilisateurs de *PUBLIC.

IBM i 21=Retrouver l'objet

Utilisez Recover object pour restaurer les objets endommagés à partir des informations stockées dans les journaux IBM MQ .

L'objet de reprise utilise la commande RCRMQMOBJ (Re-create MQ Object) pour récupérer tous les objets endommagés dans la commande. Si un objet n'est pas endommagé, aucune action n'est effectuée sur cet objet.

IBM i 22 = Enregistrement d'image

Utilisez l'image d'enregistrement pour réduire le nombre de récepteurs de journal requis pour la reprise d'un ensemble d'objets et pour réduire le temps de reprise.

La commande RCDMQMIMG prend un point de contrôle pour tous les objets sélectionnés dans la commande. Il synchronise les valeurs en cours des objets dans le système de fichiers intégré (IFS) avec des informations ultérieures sur les objets, tels que les MQPUT et les MQGET enregistrés dans les récepteurs de journal.

Une fois la commande terminée, les objets du système de fichiers IFS sont à jour et ces récepteurs de journal n'ont plus besoin d'être présents pour récupérer les objets. Tous les récepteurs de journal déconnectés peuvent être déconnectés (à condition qu'ils ne soient pas nécessaires pour récupérer d'autres objets).

IBM i Configuration de la communication pour IBM i

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour qu'elle aboutisse, il est nécessaire que la connexion soit définie et disponible.

DQM est une fonction de mise en file d'attente distante pour IBM MQ for IBM i. Il fournit des programmes de contrôle de canal pour le gestionnaire de files d'attente IBM MQ for IBM i qui forment l'interface vers les liaisons de communication, contrôlables par l'opérateur système.

Lorsqu'un canal de gestion de files d'attente réparties est démarré, il tente d'utiliser la connexion spécifiée dans la définition de canal. Pour qu'elle aboutisse, il est nécessaire que la connexion soit définie et disponible. Cette section explique comment s'assurer que la connexion est définie et disponible.

Pour qu'un canal puisse être démarré, la file d'attente de transmission doit être définie comme décrit dans cette section et doit être incluse dans la définition de canal de transmission.

Vous pouvez choisir entre les deux formes de communication suivantes entre les systèmes IBM MQ for IBM i :

- [«Définition d'une connexion TCP sous IBM i», à la page 308](#)

Pour TCP, une adresse hôte peut être utilisée et ces connexions sont configurées comme décrit dans le document *IBM i Communication Configuration Reference*.

Dans l'environnement TCP, chaque service distribué se voit attribuer une adresse TCP unique qui peut être utilisée par les machines distantes pour accéder au service. L'adresse TCP se compose d'un nom d'hôte/numéro et d'un numéro de port. Tous les gestionnaires de files d'attente utilisent un tel nombre pour communiquer entre eux via TCP.

- [«Réception sur TCP», à la page 309](#)

Cette forme de communication nécessite la définition d'un IBM i type d'unité logique SNA 6.2 (LU 6.2) qui fournit la liaison physique entre le système IBM i servant le gestionnaire de files d'attente local et le système servant le gestionnaire de files d'attente éloignées. Pour plus de détails sur la configuration des communications dans IBM i, voir *IBM i Communication Configuration Reference*.

Le cas échéant, le dispositif de déclenchement doit également être préparé avec la définition des processus et des files d'attente nécessaires.

MQ Adv. **CD** Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Tâches associées

[«Surveillance et contrôle des canaux sur IBM i», à la page 293](#)

Utilisez les commandes et les panneaux DQM pour créer, surveiller et contrôler les canaux vers les gestionnaires de files d'attente éloignées. Chaque gestionnaire de files d'attente dispose d'un programme DQM permettant de contrôler les interconnexions aux gestionnaires de files d'attente éloignées compatibles.

Référence associée

[Exemple de configuration- IBM MQ for IBM i](#)

[Exemple de planification de canal de transmission de messages pour IBM MQ for IBM i](#)

[Travaux d'intercommunication sous IBM i](#)

[Etats des canaux sous IBM i](#)

IBM i **Définition d'une connexion TCP sous IBM i**

Vous pouvez définir une connexion TCP dans la définition de canal à l'aide de la zone Nom de connexion.

La définition de canal contient une zone, CONNECTION NAME, qui contient l'adresse réseau TCP de la cible ou le nom d'hôte (par exemple ABCHOST). L'adresse du réseau TCP peut être au format IPv4 décimal à points (par exemple 127.0.0.1) ou au format IPv6 hexadécimal (par exemple 2001:DB8:0:0:0:0:0). Si le nom de connexion est un nom d'hôte ou un serveur de noms, la table d'hôtes IBM i est utilisée pour convertir le nom d'hôte en adresse d'hôte TCP.

Un numéro de port est requis pour une adresse TCP complète ; si ce numéro n'est pas fourni, le numéro de port par défaut 1414 est utilisé. Sur l'extrémité initiatrice d'une connexion (types de canal émetteur, demandeur et serveur), il est possible de fournir un numéro de port facultatif pour la connexion, par exemple:

```
Connection name 127.0.0.1 (1555)
```

Dans ce cas, l'extrémité initiatrice tente de se connecter à un programme récepteur sur le port 1555.

MQ Adv. **CD** Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement

le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Utilisation de l'option en attente du programme d'écoute TCP

Dans TCP, les connexions sont traitées comme incomplètes sauf si un établissement de liaison à trois est effectué entre le serveur et le client. Ces connexions sont appelées demandes de connexion en attente. Une valeur maximale est définie pour ces demandes de connexion en attente et peut être considérée comme un journal des demandes en attente sur le port TCP pour que le programme d'écoute accepte la demande.

Pour plus d'informations et pour connaître la valeur spécifique de IBM i, voir [«Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP sous IBM MQ for Multiplatforms»](#), à la page 290 .

Concepts associés

«Réception sur TCP», à la page 309

Les programmes de canal récepteur sont démarrés en réponse à une demande de démarrage du canal émetteur. Pour répondre à la demande de démarrage, un programme d'écoute doit être démarré pour détecter les demandes réseau entrantes et démarrer le canal associé. Vous démarrez ce programme d'écoute à l'aide de la commande STRMQMLSR.

Réception sur TCP

Les programmes de canal récepteur sont démarrés en réponse à une demande de démarrage du canal émetteur. Pour répondre à la demande de démarrage, un programme d'écoute doit être démarré pour détecter les demandes réseau entrantes et démarrer le canal associé. Vous démarrez ce programme d'écoute à l'aide de la commande STRMQMLSR.

Vous pouvez démarrer plusieurs programmes d'écoute pour chaque gestionnaire de files d'attente. Par défaut, la commande STRMQMLSR utilise le port 1414, mais vous pouvez remplacer cette valeur. Pour remplacer le paramètre par défaut, ajoutez les instructions suivantes au fichier qm.ini du gestionnaire de files d'attente sélectionné. Dans cet exemple, le programme d'écoute est requis pour utiliser le port 2500:

```
TCP:  
Port=2500
```

Le fichier qm.ini se trouve dans le répertoire IFS suivant: /QIBM/UserData/mqm/qmgrs/ *nom du gestionnaire de files d'attente*.

Cette nouvelle valeur est lue uniquement lorsque le programme d'écoute TCP est démarré. Si un programme d'écoute est déjà en cours d'exécution, cette modification n'est pas visible par ce programme. Pour utiliser la nouvelle valeur, arrêtez le programme d'écoute et relancez la commande STRMQMLSR. Désormais, chaque fois que vous utilisez la commande STRMQMLSR, le programme d'écoute utilise par défaut le nouveau port.

Vous pouvez également spécifier un numéro de port différent dans la commande STRMQMLSR. Exemple :

```
STRMQMLSR MQMNAME( queue manager name ) PORT(2500)
```

Cette modification rend le programme d'écoute par défaut sur le nouveau port pendant la durée du travail du programme d'écoute.

Utilisation de l'option TCP SO_KEEPALIVE

Si vous souhaitez utiliser l'option SO_KEEPALIVE (pour plus d'informations, voir [«Vérification de la disponibilité de l'autre extrémité du canal»](#), à la page 252) vous devez ajouter l'entrée suivante au

fichier de configuration de votre gestionnaire de files d'attente (qm.ini dans le répertoire IFS, /QIBM/UserData/mqm/qmgrs/ *nom du gestionnaire de files d'attente*):

```
TCP:
KeepAlive=yes
```

Vous devez ensuite exécuter la commande suivante:

```
CFGTCP
```

Sélectionnez l'option 3 (Modifier les attributs TCP). Vous pouvez désormais spécifier un intervalle de temps en minutes. Vous pouvez indiquer une valeur comprise entre 1 et 40320 minutes ; la valeur par défaut est 120.

Utilisation de l'option en attente du programme d'écoute TCP

Lors de la réception sur TCP, un nombre maximal de demandes de connexion en attente est défini. Ce nombre peut être considéré comme un *arriéré* de demandes en attente sur le port TCP pour que le programme d'écoute accepte la demande.

La valeur par défaut du journal des éléments en attente du programme d'écoute sous IBM i est 255. Si le journal des éléments en attente atteint cette valeur, la connexion TCP est rejetée et le canal ne peut pas démarrer.

Pour les canaux MCA, le canal passe à l'état RETRY et relance la connexion ultérieurement.

Pour les connexions client, le client reçoit un code anomalie MQRQ_Q_MGR_NOT_AVAILABLE de MQCONN et peut relancer la connexion ultérieurement.

Toutefois, pour éviter cette erreur, vous pouvez ajouter une entrée dans le fichier qm.ini :

```
ListenerBacklog = n
```

Cette valeur remplace le nombre maximal par défaut de demandes en attente (255) pour le programme d'écoute TCP.

Remarque : Certains systèmes d'exploitation prennent en charge une valeur supérieure à la valeur par défaut. Si nécessaire, cette valeur peut être utilisée pour éviter d'atteindre la limite de connexion.

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

Définissez les détails des communications LU 6.2 à l'aide d'un nom de mode, d'un nom TP et d'un nom de connexion d'une connexion LU 6.2 complète.

L'extrémité initiée de la liaison doit avoir une définition de poste de routage pour compléter cet objet CSI. Pour plus d'informations sur la gestion des demandes de travail à partir de systèmes LU 6.2 distants, voir *IBM i Programming: Work Management Guide*.

Pour plus d'informations, voir le *Guide de configuration APPC multiplateforme* et le tableau ci-après.

Plateforme distante	TPNAME
z/OS ou MVS	Identique à la valeur indiquée dans les informations complémentaires correspondantes sur le gestionnaire de files d'attente éloignées.
IBM i	Identique à la valeur de comparaison dans le poste de routage sur le système IBM i .

Tableau 26. Paramètres sur le système IBM i local pour une plateforme de gestionnaire de files d'attente éloignées (suite)

Plateforme distante	TPNAME
Systèmes AIX and Linux	Programme de transaction facturable défini dans la configuration de LU 6.2 distante.
Windows	Comme indiqué dans la commande Windows Exécuter le programme d'écoute ou dans le programme de transaction facturable qui a été défini à l'aide de TpSetup sous Windows.

Si vous disposez de plusieurs gestionnaires de files d'attente sur le même ordinateur, vérifiez que les noms de programme transactionnel dans les définitions de canal sont uniques.

Concepts associés

«Début de l'arrêt (Emetteur)», à la page 311

Utilisez la commande CRTMQMCHL pour définir un canal de type de transport *LU62.

«Fin initiée (récepteur)», à la page 314

Utilisez la commande CRTMQMCHL pour définir l'extrémité réceptrice de la liaison de canal de transmission avec le type de transport *LU62.

IBM i Début de l'arrêt (Emetteur)

Utilisez la commande CRTMQMCHL pour définir un canal de type de transport *LU62.

L'utilisation de l'objet CSI est facultative dans IBM MQ for IBM i V5.3 ou ultérieure.

Le panneau d'extrémité initiateur est illustré dans la figure [Panneau de configuration de la communication LU 6.2 -Fin initiatrice](#). Pour obtenir le panneau complet comme illustré, appuyez sur F10 à partir du premier panneau.

```

Create Comm Side Information (CRTCSI)

Type choices, press Enter.

Side information . . . . . > WINSDOA1   Name
Library . . . . . > QSYS           Name, *CURLIB
Remote location . . . . . > WINSDOA1   Name
Transaction program . . . . . > MQSERIES

Text 'description' . . . . . *BLANK

Additional Parameters

Device . . . . . *LOC           Name, *LOC
Local location . . . . . *LOC           Name, *LOC, *NETATR
Mode . . . . . JSTM0D92         Name, *NETATR
Remote network identifier . . . *LOC           Name, *LOC, *NETATR, *NONE
Authority . . . . . *LIBCRTAUT       Name, *LIBCRTAUT, *CHANGE...

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

Figure 34. Panneau de configuration de la communication LU 6.2 -Fin du lancement

Complétez les zones de fin de lancement comme suit:

Informations complémentaires

Attribuez à cette définition un nom utilisé pour stocker l'objet d'informations complémentaires à créer, par exemple, WINSDOA1.

Remarque : Pour l'unité logique 6.2, la liaison entre la définition de canal de transmission et la connexion de communication est la zone **Nom de connexion** de la définition de canal de transmission à l'extrémité émettrice. Cette zone contient le nom de l'objet CSI.

Bibliothèque

Nom de la bibliothèque dans laquelle cette définition est stockée.

L'objet CSI doit être disponible dans une bibliothèque accessible au programme servant le canal de transmission de messages, par exemple QSYS, QMQM et QGPL.

Si le nom est incorrect, manquant ou introuvable, une erreur se produit au démarrage du canal.

Lieu éloigné

Indique le nom du lieu éloigné avec lequel votre programme communique.

En résumé, ce paramètre obligatoire contient le nom d'unité logique du partenaire sur le système distant, tel que défini dans la description d'unité utilisée pour la liaison de communication entre les deux systèmes.

Vous pouvez trouver le nom du **lieu éloigné** en exécutant la commande DSPNETA sur le système éloigné et en voyant le nom du lieu local par défaut.

Programme transactionnel

Indique le nom (jusqu'à 64 caractères) du programme de transaction sur le système éloigné à démarrer. Il peut s'agir d'un nom de processus de transaction, d'un nom de programme, d'un nom de canal ou d'une chaîne de caractères qui correspond à la **valeur de comparaison** dans l'entrée de routage.

Ce paramètre est obligatoire.

Remarque : Pour indiquer des noms de programme de transaction de service SNA, entrez la représentation hexadécimale du nom de programme de transaction de service. Par exemple, pour spécifier un nom de programme de transaction de service avec une représentation hexadécimale de 21F0F0F1, entrez X'21F0F0F1'.

Pour plus d'informations sur les noms de programme de transaction de service SNA, reportez-vous au manuel *SNA Transaction Programmer's Reference* pour le type d'unité logique 6.2.

Si l'extrémité réceptrice est un autre système IBM i, le nom **Programme de transaction** est utilisé pour faire correspondre l'objet CSI à l'extrémité émettrice avec l'entrée de routage à l'extrémité réceptrice. Ce nom doit être unique pour chaque gestionnaire de files d'attente sur le système IBM i cible. Voir le paramètre **Program to call** sous *Initiated end (Receiver)*. Voir aussi le paramètre **Comparison data: compare value** dans le panneau Add Routing Entry.

Texte descriptif

Description (jusqu'à 50 caractères) pour vous rappeler l'utilisation prévue de cette connexion.

Dispositif

Indique le nom de la description d'unité utilisée pour le système éloigné. Les valeurs possibles sont les suivantes:

***LOC**

L'unité est déterminée par le système.

Nom de l'unité

Indiquez le nom de l'unité associée au lieu éloigné.

Emplacement local

Indique le nom du lieu local. Les valeurs possibles sont les suivantes:

***LOC**

Le nom du lieu local est déterminé par le système.

***NETATR**

La valeur LCLLOCNAME indiquée dans les attributs de réseau du système est utilisée.

Nom-lieu-local

Indiquez le nom de votre emplacement. Indiquez le lieu local si vous souhaitez indiquer un nom de lieu spécifique pour le lieu éloigné. Le nom du lieu peut être trouvé à l'aide de la commande DSPNETA.

Mode

Indique le mode utilisé pour contrôler la session. Ce nom est identique à CPI (Common Programming Interface)-Communications Mode_Name. Les valeurs possibles sont les suivantes:

***NETATR**

Le mode des attributs de réseau est utilisé.

VIDE

Huit caractères blancs sont utilisés.

Nom-mode

Indiquez un nom de mode pour le lieu éloigné.

Remarque : Etant donné que le mode détermine la priorité de transmission de la session de communication, il peut être utile de définir différents modes en fonction de la priorité des messages envoyés ; par exemple, MQMODE_HI, MQMODE_MED et MQMODE_LOW. (Vous pouvez avoir plusieurs CSI pointant vers le même emplacement.)

ID éloigné du réseau

Indique l'ID éloigné de réseau utilisé avec le lieu éloigné. Les valeurs possibles sont les suivantes:

***LOC**

L'ID réseau éloigné du lieu éloigné est utilisé.

***NETATR**

L'ID éloigné de réseau indiqué dans les attributs de réseau est utilisé.

***NONE**

Le réseau distant n'a pas de nom.

ID-réseau-distant

Indiquez un ID éloigné de réseau. Utilisez la commande DSPNETA sur le lieu éloigné pour rechercher le nom de cet ID réseau. Il s'agit de l'ID réseau local sur le lieu éloigné.

Droits d'accès

Indique les droits que vous octroyez aux utilisateurs qui ne disposent pas de droits spécifiques sur l'objet, qui ne figurent pas dans une liste d'autorisation et dont le profil de groupe ne dispose pas de droits spécifiques sur l'objet. Les valeurs possibles sont les suivantes:

***LIBCRTAUT**

Les droits publics sur l'objet proviennent du paramètre CRTAUT de la bibliothèque indiquée. Cette valeur est déterminée au moment de la création. Si la valeur CRTAUT de la bibliothèque est modifiée après la création de l'objet, la nouvelle valeur n'affecte pas les objets existants.

***CHANGE**

Le droit de modification permet à l'utilisateur d'exécuter des fonctions de base sur l'objet, mais l'utilisateur ne peut pas modifier l'objet. Le droit de modification fournit le droit d'opération sur l'objet et tous les droits sur les données.

***ALL**

L'utilisateur peut effectuer toutes les opérations, à l'exception des opérations limitées au propriétaire ou contrôlées par le droit de gestion des listes d'autorisation. L'utilisateur peut contrôler l'existence de l'objet et spécifier la sécurité de l'objet, modifier l'objet et exécuter des fonctions de base sur l'objet. L'utilisateur peut modifier la propriété de l'objet.

***USE**

Le droit d'utilisation fournit le droit d'opération sur les objets et le droit de lecture.

***EXCLUDE**

Le droit d'exclusion empêche l'utilisateur d'accéder à l'objet.

Liste d'autorisation

Indiquez le nom de la liste d'autorisation avec les droits utilisés pour les informations complémentaires.

IBM i Fin initiée (récepteur)

Utilisez la commande CRTMQMCHL pour définir l'extrémité réceptrice de la liaison de canal de transmission avec le type de transport *LU62.

Laissez la zone CONNECTION NAME vide et vérifiez que les détails correspondants correspondent à l'extrémité émettrice du canal. Pour plus de détails, voir [Création d'un canal](#).

Pour permettre à l'extrémité initiatrice de démarrer le canal récepteur, ajoutez un poste de routage à un sous-système à l'extrémité initiée. Le sous-système doit être celui qui alloue l'unité APPC utilisée dans les sessions LU 6.2. Par conséquent, il doit disposer d'un poste de communication valide pour cette unité. Le poste de routage appelle le programme qui démarre l'extrémité réceptrice du canal de transmission de messages.

Utilisez les commandes IBM i (par exemple, ADDRTGE) pour définir l'extrémité de la liaison initiée par une session de communication.

Le panneau d'extrémité initié apparaît dans le panneau de configuration de la communication [LU 6.2 -ajout d'un poste de routage](#).

```
Add Routing Entry (ADDRTGE)
```

```
Type choices, press Enter.
```

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

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

```
Bottom
```

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

Figure 35. LU 6.2 panneau de configuration des communications-fin initiée

Description du sous-système

Nom du sous-système dans lequel réside cette définition. Utilisez la commande IBM i WRKSBSD pour afficher et mettre à jour la description de sous-système appropriée pour le poste de routage.

Numéro de séquence du poste de routage

Numéro unique dans votre sous-système pour identifier cette définition de communication. Vous pouvez utiliser des valeurs comprises entre 1 et 9999.

Données de comparaison: valeur de comparaison

Chaîne de texte à comparer à la chaîne reçue lorsque la session est démarrée par un paramètre

Programme de transaction, comme illustré dans la [Figure 1](#). La chaîne de caractères est dérivée de la zone Programme de transaction de l'inventaire CSI de l'expéditeur.

Données de comparaison: position de départ

Position du caractère dans la chaîne où la comparaison doit commencer.

Remarque : La zone de position de début correspond à la position de caractère dans la chaîne à des fins de comparaison, et cette position est toujours 37.

Programme à appeler

Nom du programme qui exécute le programme de messages entrants à appeler pour démarrer la session.

Le programme AMQCRC6A est appelé pour le gestionnaire de files d'attente par défaut. Ce programme est fourni avec IBM MQ for IBM i et configure l'environnement, puis appelle AMQCRS6A.

Pour les gestionnaires de files d'attente supplémentaires:

- Chaque gestionnaire de files d'attente possède un programme d'unité logique 6.2 facturable spécifique situé dans sa bibliothèque. Ce programme est appelé AMQCRC6B et est automatiquement généré lors de la création du gestionnaire de files d'attente.
- Chaque gestionnaire de files d'attente requiert un poste de routage spécifique avec des données de routage uniques à ajouter. Ces données de routage doivent correspondre au nom du **programme de transaction** fourni par le système demandeur (voir [Initiating end \(Sender\)](#)).

Voici un exemple: [LU 6.2 communication setup panel-display routing entries](#):

```
Display Routing Entries
System: MY400
Subsystem description: QCMN      Status: ACTIVE

Type options, press Enter.
5=Display details

Start
Opt  Seq Nbr  Program      Library      Compare Value  Pos
10   *RTGDTA   'QZSCSRVR'   'QZSCSRVR'   37
20   *RTGDTA   'QZRCSRVR'   'QZRCSRVR'   37
30   *RTGDTA   'QZHQTRG'    'QZHQTRG'    37
50   *RTGDTA   'QVPPRINT'   'QVPPRINT'   37
60   *RTGDTA   'QNPSERV'    'QNPSERV'    37
70   *RTGDTA   'QNMAPINGD'  'QNMAPINGD'  37
80   QNMAREXECD QSYS        'AREXECD'    37
90   AMQCRC6A  QMQMBW     'MQSERIES'   37
100  *RTGDTA   'QTFDWNLD'   'QTFDWNLD'   37
150  *RTGDTA   'QMFRCVR'    'QMFRCVR'    37

F3=Exit  F9=Display all detailed descriptions  F12=Cancel
```

Figure 36. LU 6.2 panneau de configuration des communications-fin initiée

Dans le panneau de configuration des communications [LU 6.2 -affichage des entrées de routage](#), le numéro de séquence 90 représente le gestionnaire de files d'attente par défaut et fournit la compatibilité avec les configurations des éditions précédentes (c'est-à-dire V3R2, V3R6, V3R7 et V4R2) de IBM MQ for IBM i. Ces éditions n'autorisent qu'un seul gestionnaire de files d'attente. Les numéros de séquence 92 et 94 représentent deux gestionnaires de files d'attente supplémentaires appelés ALPHA et BETA qui sont créés avec les bibliothèques QMALPHA et QMBETA.

Remarque : Vous pouvez avoir plusieurs postes de routage pour chaque gestionnaire de files d'attente en utilisant des données de routage différentes. Ces entrées offrent l'option de priorités de travail différentes en fonction des classes utilisées.

Classe

Nom et bibliothèque de la classe utilisée pour les étapes démarrées via ce poste de routage. La classe définit les attributs de l'environnement d'exécution de la phase d'activation et indique la priorité du travail. Vous devez indiquer une entrée de classe appropriée. Utilisez, par exemple, la commande WRKCLS pour afficher les classes existantes ou pour créer une classe. Pour plus d'informations sur la gestion des demandes de travail à partir de systèmes LU 6.2 distants, voir *IBM i Programming: Work Management Guide*.

Remarque sur la gestion du travail

Le travail AMQCRS6A ne peut pas tirer parti des fonctions de gestion des travaux IBM i normales décrites dans la rubrique [Gestion des travaux](#) car il n'est pas démarré de la même manière que les autres travaux IBM MQ . Pour modifier les propriétés d'exécution des travaux du récepteur LU62 , vous pouvez effectuer l'une des modifications suivantes:

- Modifier la description de classe spécifiée sur l'entrée de routage pour le travail AMQCRS6A
- Modification de la description de travail sur le poste communications

Pour plus d'informations sur la configuration des travaux de communication, voir *IBM i Programming: Work Management Guide* .

Configuration d'un cluster de gestionnaires de files d'attente

Les clusters fournissent un mécanisme d'interconnexion des gestionnaires de files d'attente qui simplifie à la fois la configuration initiale et la gestion en cours. Vous pouvez définir des composants de cluster et créer et gérer des clusters.

Avant de commencer

Pour une introduction aux concepts de classification, voir [Clusters](#).

Lorsque vous concevez votre cluster de gestionnaires de files d'attente, vous devez prendre des décisions. Voir [Exemples de clusters](#) et [Conception de clusters](#).

Tâches associées

«[Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente](#)», à la page 476

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

Référence associée

[Supprimer la rubrique](#)

Définition des composants d'un cluster

Les clusters sont composés de gestionnaires de files d'attente, de canaux de cluster et de files d'attente de cluster. Vous pouvez définir des files d'attente de cluster et modifier certains aspects des objets de cluster par défaut. Vous pouvez obtenir des informations de configuration et de statut sur les canaux définis automatiquement et sur la relation entre les canaux émetteurs de cluster individuels et les files d'attente de transmission.

Pour plus d'informations sur la définition de chacun des composants de cluster, voir les sous-rubriques suivantes:

Concepts associés

[Composants d'un cluster](#)

[Canaux de cluster](#)

Tâches associées

[Définition de rubriques de cluster](#)

«[Configuration d'un nouveau cluster](#)», à la page 331

Suivez ces instructions pour configurer l'exemple de cluster. Des instructions distinctes décrivent la configuration du cluster sur TCP/IP, LU 6.2 et avec une ou plusieurs files d'attente de transmission. Testez le cluster en envoyant un message d'un gestionnaire de files d'attente à l'autre.

«[Ajout d'un gestionnaire de files d'attente à un cluster](#)», à la page 343

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages vers les files d'attente de cluster et les rubriques sont transférés à l'aide de la file d'attente de transmission de cluster unique SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Définition de files d'attente de cluster

Une file d'attente de cluster est une file d'attente hébergée par un gestionnaire de files d'attente de cluster et accessible aux autres gestionnaires de files d'attente dans le cluster. Définissez une file d'attente de cluster en tant que file d'attente locale sur le gestionnaire de files d'attente de cluster où la file d'attente est hébergée. Indiquez le nom du cluster auquel appartient la file d'attente.

L'exemple suivant illustre une commande **runmqsc** permettant de définir une file d'attente de cluster avec l'option CLUSTER :

```
DEFINE QLOCAL(Q1) CLUSTER(SALES)
```

Une file d'attente de cluster est annoncée aux autres gestionnaires de files d'attente de cluster. Ces autres gestionnaires de files d'attente de cluster peuvent insérer des messages dans une file d'attente de cluster sans qu'une définition de file d'attente éloignée correspondante soit nécessaire. Une file d'attente de cluster peut être annoncée dans plusieurs clusters à l'aide d'une liste de noms de cluster.

Lorsqu'une file d'attente est annoncée, les gestionnaires de files d'attente de cluster peuvent y insérer des messages. Pour insérer un message, le gestionnaire de files d'attente doit déterminer, à partir des référentiels complets, l'emplacement où la file d'attente est hébergée. Il ajoute alors des informations de routage au message et insère ce dernier dans une file d'attente de transmission du cluster.

Une file d'attente de cluster peut être une file d'attente partagée par les membres d'un groupe de partage de files d'attente dans IBM MQ for z/OS.

Liaison en cours

Vous pouvez créer un cluster dans lequel plusieurs gestionnaires de files d'attente hébergent une instance de la même file d'attente de cluster. Assurez-vous que tous les messages d'une séquence sont envoyés à la même instance de la file d'attente. Vous pouvez lier une série de messages à une file d'attente particulière à l'aide de l'option MQOO_BIND_ON_OPEN de l'appel MQOPEN .

Files d'attente de transmission de cluster

Un gestionnaire de files d'attente peut stocker des messages pour d'autres gestionnaires d'un cluster dans plusieurs files d'attente de transmission. Vous pouvez configurer un gestionnaire de files d'attente de deux manières pour stocker les messages dans plusieurs files d'attente de transmission de cluster. Si vous définissez l'attribut de gestionnaire de files d'attente **DEFCLXQ** sur CHANNEL, une file d'attente de transmission de cluster différente est créée automatiquement à partir de SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE pour chaque canal émetteur de cluster. Si vous définissez l'option de file d'attente de transmission CLCHNAME sur un ou plusieurs canaux émetteurs de cluster, le gestionnaire de files d'attente peut stocker les messages pour les canaux correspondants dans cette file d'attente de transmission.



Avertissement : Si vous utilisez SYSTEM . CLUSTER . TRANSMIT . QUEUES avec un gestionnaire de files d'attente mis à niveau à partir d'une version du produit antérieure à IBM WebSphere MQ 7.5, vérifiez que SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE a l'option SHARE/NOSHARE définie sur **SHARE**.

Un message pour une file d'attente de cluster sur un gestionnaire de files d'attente différent est placé sur une file d'attente de transmission de cluster avant d'être envoyé. Un canal émetteur de cluster transfère les messages d'une file d'attente de transmission de cluster vers des canaux récepteurs de cluster sur d'autres gestionnaires de files d'attente. Par défaut, une file d'attente de transmission de cluster définie par le système contient tous les messages qui doivent être transférés vers d'autres gestionnaires de files d'attente de cluster. La file d'attente est appelée SYSTEM . CLUSTER . TRANSMIT . QUEUE. Un gestionnaire de files d'attente faisant partie d'un cluster peut envoyer des messages sur cette file d'attente de transmission de cluster à n'importe quel autre gestionnaire de files d'attente du même cluster.

Une définition pour la file d'attente SYSTEM . CLUSTER . TRANSMIT . QUEUE unique est créée par défaut sur chaque gestionnaire de files d'attente Multiplatforms. Sur z/OS, la définition peut être définie à l'aide de l'exemple **CSQ4INSX**fourni.

Vous pouvez configurer un gestionnaire de files d'attente pour le transfert de messages vers d'autres gestionnaires de files d'attente en cluster à l'aide de plusieurs files d'attente de transmission. Vous pouvez définir des files d'attente de transmission de cluster supplémentaires manuellement ou demander au gestionnaire de files d'attente de les créer automatiquement.

Pour que les files d'attente soient créées automatiquement par le gestionnaire de files d'attente, remplacez l'attribut de gestionnaire de files d'attente DEFCLXQ SCTQ par CHANNEL. Le résultat est que le gestionnaire de files d'attente crée une file d'attente de transmission de cluster individuelle pour chaque canal émetteur de cluster créé. Les files d'attente de transmission sont créées en tant que files d'attente dynamiques permanentes à partir de la file d'attente modèle, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. Le nom de chaque file d'attente dynamique permanente est SYSTEM . CLUSTER . TRANSMIT . *ChannelName*. Le nom du canal émetteur de cluster auquel chaque file d'attente de transmission de cluster dynamique permanente est associée est défini dans l'attribut de file d'attente de transmission locale CLCHNAME. Les messages des gestionnaires de files d'attente en cluster éloignées sont placés dans la file d'attente de transmission de cluster dynamique permanente pour le canal émetteur de cluster associé, plutôt que dans SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Pour créer les files d'attente de transmission de cluster manuellement, créez une file d'attente locale avec l'attribut USAGE défini sur XMITQet l'attribut CLCHNAME défini sur un nom de canal générique qui se résout en un ou plusieurs canaux émetteurs de cluster ; voir [ClusterChannelName](#). Si vous créez des files d'attente de transmission de cluster manuellement, vous avez le choix d'associer la file d'attente de transmission à un canal émetteur de cluster unique ou à plusieurs canaux émetteurs de cluster. L'attribut CLCHNAME est un nom générique, ce qui signifie que vous pouvez placer plusieurs caractères génériques, "*", dans le nom.

A l'exception des canaux émetteurs de cluster initiaux que vous créez manuellement pour connecter un gestionnaire de files d'attente à un référentiel complet, les canaux émetteurs de cluster sont créés automatiquement. Ils sont créés automatiquement lorsqu'il y a un message à transférer vers un gestionnaire de files d'attente de cluster. Ils sont créés avec le même nom que le nom du canal récepteur de cluster qui reçoit les messages de cluster pour ce cluster particulier sur le gestionnaire de files d'attente de destination.

Si vous suivez une convention de dénomination pour les canaux récepteurs de cluster, il est possible de définir une valeur générique pour CLCHNAME qui filtre différents types de messages de cluster dans différentes files d'attente de transmission. Par exemple, si vous respectez la convention de dénomination des canaux récepteurs de cluster de *ClusterName . QmgrName*, le nom générique *ClusterName . ** filtre les messages des différents clusters sur des files d'attente de transmission différentes. Vous devez définir les files d'attente de transmission manuellement et définir CLCHNAME dans chaque file d'attente de transmission sur *ClusterName . **.

Les modifications apportées à l'association des files d'attente de transmission de cluster aux canaux émetteurs de cluster ne prennent pas effet immédiatement. La file d'attente de transmission actuellement associée qu'un canal émetteur de cluster traite peut contenir des messages qui sont en cours de transfert par le canal émetteur de cluster. Ce n'est que lorsqu'aucun message de la file d'attente de transmission actuellement associée n'est traité par un canal émetteur de cluster que le gestionnaire de files d'attente peut modifier l'association du canal émetteur de cluster à une file d'attente de transmission différente. Cela peut se produire lorsqu'il ne reste aucun message dans la file d'attente de transmission à traiter par le canal émetteur de cluster ou lorsque le traitement des messages est interrompu et que le canal émetteur de cluster ne comporte aucun message "en cours". Lorsque cela se produit, les messages non traités pour le canal émetteur de cluster sont transférés dans la file d'attente de transmission qui vient d'être associée et l'association du canal émetteur de cluster est modifiée.

Vous pouvez créer une définition de file d'attente éloignée qui se résout en file d'attente de transmission de cluster. Dans la définition, le gestionnaire de files d'attente QMX se trouve dans le même cluster que le gestionnaire de files d'attente local et il n'existe pas de file d'attente de transmission, QMX.

```
DEFINE QREMOTE(A) RNAME(B) RQMNAME(QMX)
```

Lors de la résolution de nom de file d'attente, la file d'attente de transmission du cluster est prioritaire sur la file d'attente de transmission par défaut. Un message inséré dans A est stocké dans la file d'attente de transmission du cluster, puis envoyé à la file d'attente éloignée B sur QMX.

Les gestionnaires de files d'attente peuvent également communiquer avec d'autres gestionnaires de files d'attente qui ne font pas partie d'un cluster. Vous devez définir des canaux et une file d'attente de transmission vers l'autre gestionnaire de files d'attente, de la même manière que dans un environnement de files d'attente réparties.

Remarque : Les applications doivent écrire dans des files d'attente qui se résolvent dans la file d'attente de transmission du cluster et ne doivent pas écrire directement dans la file d'attente de transmission du cluster.

Définition automatique des files d'attente éloignées

Un gestionnaire de files d'attente dans un cluster n'a pas besoin d'une définition de file d'attente éloignée pour les files d'attente éloignées du cluster. Le gestionnaire de files d'attente de cluster trouve l'emplacement d'une file d'attente éloignée dans le référentiel complet. Il ajoute des informations de routage au message et les place dans la file d'attente de transmission du cluster. IBM MQ crée automatiquement une définition équivalente à une définition de file d'attente éloignée afin que le message puisse être envoyé.

Vous ne pouvez pas modifier ou supprimer une définition de file d'attente éloignée créée automatiquement. Toutefois, en utilisant la commande `DISPLAY QUEUE runmqsc` avec l'attribut `CLUSINFO`, vous pouvez afficher toutes les files d'attente locales sur un gestionnaire de files d'attente ainsi que toutes les files d'attente de cluster, y compris les files d'attente de cluster sur les gestionnaires de files d'attente éloignées. Exemple :

```
DISPLAY QUEUE(*) CLUSINFO
```

Concepts associés

[Files d'attente de cluster](#)

[Comment choisir le type de file d'attente de transmission de cluster à utiliser](#)

Référence associée

[ClusterChannelNom \(MQCHAR20\)](#)

Utilisation des canaux émetteurs de cluster définis automatiquement

Après avoir introduit un gestionnaire de files d'attente dans un cluster en créant ses définitions `CLUSDR` et `CLUSRCVR` initiales, IBM MQ crée automatiquement d'autres définitions de canal émetteur de cluster lorsque cela est nécessaire pour déplacer des messages vers un autre gestionnaire de files d'attente dans le cluster. Vous pouvez afficher des informations sur les canaux émetteurs de cluster définis automatiquement, mais vous ne pouvez pas les modifier. Pour modifier leur comportement, vous pouvez utiliser un exit de définition automatique de canal.

Avant de commencer

Pour une introduction aux canaux définis automatiquement, voir [Canaux émetteurs de cluster définis automatiquement](#).

Pourquoi et quand exécuter cette tâche

Les canaux émetteurs de cluster définis automatiquement sont créés par le cluster au fur et à mesure des besoins, et ils restent actifs jusqu'à ce qu'ils soient arrêtés à l'aide des règles d'intervalle de déconnexion normales.

Les canaux émetteurs de cluster (CLUSDRs) peuvent être définis automatiquement pour déplacer des messages d'application et des messages d'administration de cluster internes. Par exemple, dans un cluster de publication / abonnement (dans lequel une rubrique en cluster a été définie), des canaux peuvent être définis entre des référentiels partiels pour permettre l'échange de l'état d'abonnement de proxy'. Lorsqu'elles ne sont pas requises (inactives) pendant une période prolongée, les CLUSSDR définies automatiquement sont supprimées du cache d'informations de cluster d'un référentiel partiel et ne sont plus visibles sur ce gestionnaire de files d'attente.

Multi Sous Multiplateformes, le gestionnaire des droits d'accès aux objets (OAM) ne connaît pas l'existence de canaux émetteurs de cluster définis automatiquement. Si vous émettez des commandes **start**, **stop**, **ping**, **reset** ou **resolve** sur un canal émetteur de cluster défini automatiquement, la méthode d'accès aux objets (OAM) vérifie si vous êtes autorisé à effectuer la même action sur le canal récepteur de cluster correspondant.

z/OS Sous z/OS, vous pouvez sécuriser un canal émetteur de cluster défini automatiquement de la même manière que n'importe quel autre canal.

Procédure

- Affiche des informations sur les canaux définis automatiquement pour un gestionnaire de files d'attente de cluster donné.

Vous ne pouvez pas voir les canaux définis automatiquement à l'aide de la commande DISPLAY CHANNEL **runmqsc** . Pour afficher les canaux définis automatiquement, utilisez la commande suivante:

```
DISPLAY CLUSQMGR(qMgrName)
```

- Affiche l'état du canal défini automatiquement pour un CLUSRCVR donné.

Pour afficher le statut du canal CLUSSDR défini automatiquement correspondant à une définition de canal CLUSRCVR que vous avez créée, utilisez la commande suivante:

```
DISPLAY CHSTATUS(channelname)
```

- Utilisez un exit de définition automatique de canal pour modifier le comportement d'un canal défini automatiquement.

Vous pouvez utiliser l'exit de définition automatique de canal IBM MQ si vous souhaitez écrire un programme d'exit utilisateur pour personnaliser un canal émetteur de cluster ou un canal récepteur de cluster. Par exemple, vous pouvez utiliser l'exit de définition automatique de canal dans un environnement de cluster pour effectuer l'une des modifications suivantes:

- Personnalisez les définitions de communication, c'est-à-dire les noms SNA LU6.2 .
- Ajoutez ou supprimez d'autres exits, par exemple des exits de sécurité.
- Modifiez les noms des exits de canal.

Le nom de l'exit de canal CLUSSDR est généré automatiquement à partir de la définition de canal CLUSRCVR et peut donc ne pas être adapté à vos besoins, en particulier si les deux extrémités du canal se trouvent sur des plateformes différentes.

Le format des noms d'exit est différent sur les différentes plateformes. Exemple :

- **z/OS** Sur la plateforme z/OS , le format du paramètre SCYEXIT (*nom de l'exit de sécurité*) est SCYEXIT (' SECEXIT ')

- **Windows** Sur les plateformes Windows , le format du paramètre SCYEXIT (*nom de l'exit de sécurité*) est SCYEXIT(' *drive:\path\library (secexit)'*)

Remarque : **z/OS** S'il n'existe pas d'exit de définition automatique de canal, le gestionnaire de files d'attente z/OS dérive le nom d'exit de canal CLUSSDR de la définition de canal CLUSRCVR à l'autre extrémité du canal. Pour dériver le nom de l'exit z/OS d'un nom nonz/OS , l'algorithme suivant est utilisé:

- Les noms d'exit sous Multiplateformes sont au format général *path/library (fonction)*.
- Si *fonction* est présente, jusqu'à huit caractères sont utilisés.
- Sinon, jusqu'à huit caractères de la *bibliothèque* sont utilisés.

Exemple :

- /var/mqm/exits/myExit.so(MsgExit) convertit en MSGEXIT
- /var/mqm/exits/myExit convertit en MYEXIT
- /var/mqm/exits/myExit.so(ExitLongName) convertit en EXITLONG
- Si le cluster doit utiliser **PROPCTL** pour supprimer des en-têtes d'application tels que RFH2 des messages d'un gestionnaire de files d'attente IBM MQ vers un gestionnaire de files d'attente d'une version antérieure du produit, vous devez écrire un exit de définition automatique de canal qui affecte à **PROPCTL** la valeur NONE.
- Utilisez l'attribut de canal LOCLADDR pour contrôler les aspects de l'adressage.
 - Pour permettre à un canal sortant (TCP) d'utiliser une adresse IP, un port ou une plage de ports spécifique, utilisez l'attribut de canal LOCLADDR. Cela est utile si vous disposez de plusieurs cartes réseau et que vous souhaitez qu'un canal utilise une carte spécifique pour les communications sortantes.
 - Pour spécifier une adresse IP virtuelle sur les canaux CLUSSDR , utilisez l'adresse IP de LOCLADDR sur un CLUSSDR défini manuellement. Pour spécifier la plage de ports, utilisez la plage de ports de CLUSRCVR.
 - Si un cluster doit utiliser LOCLADDR pour obtenir les canaux de communication sortants à lier à une adresse IP spécifique, vous pouvez écrire un exit de définition automatique de canal pour forcer la valeur LOCLADDR dans l'un de leurs canaux CLUSSDR automatiquement définis. Vous devez également le spécifier dans le canal CLUSSDR défini manuellement.
 - Placez un numéro de port ou une plage de ports dans LOCLADDR d'un canal CLUSRCVR , si vous souhaitez que tous les gestionnaires de files d'attente d'un cluster utilisent un port ou une plage de ports spécifique pour toutes leurs communications sortantes.

Remarque : N'insérez pas d'adresse IP dans la zone LOCLADDR d'un canal CLUSRCVR , sauf si tous les gestionnaires de files d'attente se trouvent sur le même serveur. L'adresse IP LOCLADDR est propagée aux canaux CLUSSDR définis automatiquement de tous les gestionnaires de files d'attente qui se connectent à l'aide du canal CLUSRCVR .

Multi Sous Multiplateformes, vous pouvez définir une valeur d'adresse locale par défaut qui est utilisée pour tous les canaux émetteurs pour lesquels aucune adresse locale n'est définie. La valeur par défaut est définie en définissant la variable d'environnement MQ_LCLADDR avant de démarrer le gestionnaire de files d'attente. Le format de la valeur correspond à celui de l'attribut MQSC LOCLADDR.

Référence associée

Adresse locale (LOCLADDR)

Utilisation des objets de cluster par défaut

Vous pouvez modifier les définitions de canal par défaut de la même manière que toute autre définition de canal, en exécutant des commandes MQSC ou PCF. Ne modifiez pas les définitions de files d'attente par défaut, sauf SYSTEM . CLUSTER . HISTORY . QUEUE.

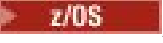
Pour obtenir la liste complète de ces objets, voir Objets de cluster par défaut. La liste suivante inclut uniquement les objets que vous pouvez modifier.

SYSTEM.CLUSTER.HISTORY.QUEUE

Chaque gestionnaire de files d'attente d'un cluster dispose d'une file d'attente locale appelée SYSTEM.CLUSTER.HISTORY.QUEUE. SYSTEM.CLUSTER.HISTORY.QUEUE est utilisé pour stocker l'historique des informations d'état de cluster à des fins de service.

Dans les paramètres d'objet par défaut, SYSTEM.CLUSTER.HISTORY.QUEUE est défini sur PUT (ENABLED). Pour supprimer la collecte d'historique, remplacez le paramètre par PUT (DISABLED).

SYSTEM.CLUSTER.TRANSMIT.QUEUE

Chaque gestionnaire de files d'attente dispose d'une file d'attente locale appelée SYSTEM.CLUSTER.TRANSMIT.QUEUE. SYSTEM.CLUSTER.TRANSMIT.QUEUE est la file d'attente de transmission par défaut pour tous les messages vers toutes les files d'attente et tous les gestionnaires de files d'attente se trouvant dans des clusters. Vous pouvez modifier la file d'attente de transmission par défaut de chaque canal émetteur de cluster en SYSTEM.CLUSTER.TRANSMIT.Channe1Name, en modifiant l'attribut de gestionnaire de files d'attente DEFXXMITQ , sauf sur z/OS. Vous ne pouvez pas supprimer SYSTEM.CLUSTER.TRANSMIT.QUEUE. Il est également utilisé pour définir les vérifications d'autorisation si la file d'attente de transmission par défaut utilisée est SYSTEM.CLUSTER.TRANSMIT.QUEUE ou SYSTEM.CLUSTER.TRANSMIT.Channe1Name.

Concepts associés

[Objets de cluster par défaut](#)

Utilisation de files d'attente de transmission de cluster et de canaux émetteurs de cluster

Les messages entre les gestionnaires de files d'attente de cluster sont stockés dans des files d'attente de transmission de cluster et transmis par les canaux émetteurs de cluster. A tout moment, un canal émetteur de cluster est associé à une file d'attente de transmission. Si vous modifiez la configuration du canal, il se peut qu'il passe à une autre file d'attente de transmission lors de son prochain démarrage. Le traitement de ce commutateur est automatisé et transactionnel.

Exécutez la commande MQSC suivante pour afficher les files d'attente de transmission auxquelles les canaux émetteurs de cluster sont associés:

```
DISPLAY CHSTATUS(*) WHERE(CHLTYPE EQ CLUSSDR)
```

```
AMQ8417: Display Channel Status details.  
CHANNEL (TO.QM2)          CHLTYPE (CLUSSDR)  
CONNAME (9.146.163.190(1416))  CURRENT  
RQMNAME (QM2)             STATUS (STOPPED)  
SUBSTATE ( )              XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

La file d'attente de transmission affichée dans le statut de canal sauvegardé d'un canal émetteur de cluster arrêté peut changer lorsque le canal redémarre. [«Sélection des files d'attente de transmission par défaut par les canaux émetteurs de cluster»](#), à la page 323 décrit le processus de sélection d'une file d'attente de transmission par défaut ; [«Sélection des files d'attente de transmission définies manuellement par les canaux émetteurs de cluster»](#), à la page 324 décrit le processus de sélection d'une file d'attente de transmission définie manuellement.

Lorsqu'un canal émetteur de cluster démarre, il revérifie son association avec les files d'attente de transmission. Si la configuration des files d'attente de transmission ou les valeurs par défaut du gestionnaire de files d'attente changent, le canal peut être réassocié à une autre file d'attente de transmission. Si le canal redémarre avec une file d'attente de transmission différente suite à un changement de configuration, un processus de transfert de messages vers la file d'attente de transmission nouvellement associée a lieu. Le [«Fonctionnement du processus de basculement du canal émetteur de cluster vers une autre file d'attente de transmission»](#), à la page 325 décrit le processus de transfert d'un canal émetteur de cluster d'une file d'attente de transmission à une autre.

Le comportement des canaux émetteurs de cluster est différent des canaux émetteurs et serveurs. Ils restent associés à la même file d'attente de transmission jusqu'à ce que l'attribut de canal **XMITQ** soit

modifié. Si vous modifiez l'attribut de file d'attente de transmission sur un canal émetteur ou serveur et que vous le redémarrez, les messages ne sont pas transférés de l'ancienne file d'attente de transmission vers la nouvelle.

Une autre différence entre les canaux émetteurs de cluster et les canaux émetteurs ou serveurs est que plusieurs canaux émetteurs de cluster peuvent ouvrir une file d'attente de transmission de cluster, mais qu'un seul canal émetteur ou serveur peut ouvrir une file d'attente de transmission normale. Vous avez la possibilité de ne pas partager les files d'attente de transmission avec les canaux émetteurs de cluster. L'exclusivité n'est pas appliquée ; il s'agit d'un résultat de la configuration. Vous pouvez configurer le chemin d'accès d'un message dans un cluster afin qu'il ne partage pas de files d'attente de transmission ou de canaux avec des messages qui circulent entre d'autres applications. Voir [Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster](#) et «Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 379.

Sélection des files d'attente de transmission par défaut par les canaux émetteurs de cluster

Une file d'attente de transmission de cluster est soit une file d'attente par défaut du système, dont le nom commence par SYSTEM.CLUSTER.TRANSMIT, soit une file d'attente définie manuellement. Un canal émetteur de cluster est associé à une file d'attente de transmission de cluster de l'une des deux manières suivantes: par le mécanisme de file d'attente de transmission de cluster par défaut ou par une configuration manuelle.

La file d'attente de transmission de cluster par défaut est définie en tant qu'attribut de gestionnaire de files d'attente, **DEFCLXQ**. Sa valeur est SCTQ ou CHANNEL. Les gestionnaires de files d'attente nouveaux et migrés sont définis sur SCTQ. Vous pouvez modifier la valeur en CHANNEL.

Si SCTQ est défini, la file d'attente de transmission de cluster par défaut est SYSTEM.CLUSTER.TRANSMIT.QUEUE. Chaque canal émetteur de cluster peut ouvrir cette file d'attente. Les canaux émetteurs de cluster qui ouvrent la file d'attente sont ceux qui ne sont pas associés à des files d'attente de transmission de cluster définies manuellement.

Si CHANNEL est défini, le gestionnaire de files d'attente peut créer une file d'attente de transmission dynamique permanente distincte pour chaque canal émetteur de cluster. Chaque file d'attente est nommée SYSTEM.CLUSTER.TRANSMIT.ChannelName et est créée à partir de la file d'attente modèle, SYSTEM.CLUSTER.TRANSMIT.MODEL.QUEUE. Chaque canal émetteur de cluster qui n'est pas associé à une file d'attente de transmission de cluster définie manuellement est associé à une file d'attente de transmission de cluster dynamique permanente. La file d'attente est créée par le gestionnaire de files d'attente lorsqu'elle requiert une file d'attente de transmission de cluster distincte pour la destination de cluster prise en charge par ce canal émetteur de cluster et qu'aucune file d'attente n'existe.

Certaines destinations de cluster peuvent être servies par des canaux émetteurs de cluster associés à des files d'attente de transmission définies manuellement, et d'autres par la ou les files d'attente par défaut. Dans l'association des canaux émetteurs de cluster avec les files d'attente de transmission, les files d'attente de transmission définies manuellement ont toujours la priorité sur les files d'attente de transmission par défaut.

La priorité des files d'attente de transmission de cluster est illustrée dans [Figure 37](#), à la page 324. Le seul canal émetteur de cluster non associé à une file d'attente de transmission de cluster définie manuellement est CS.QM1. Elle n'est pas associée à une file d'attente de transmission définie manuellement, car aucun des noms de canal de l'attribut **CLCHNAME** des files d'attente de transmission ne correspond à CS.QM1.

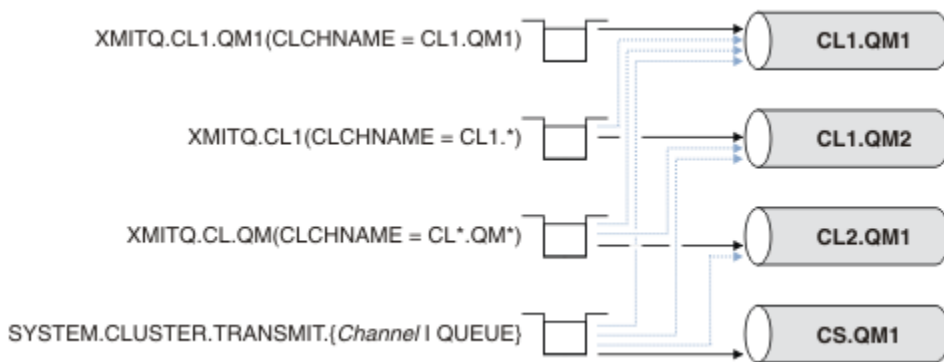


Figure 37. Priorité de la file d'attente/du canal émetteur de cluster

Sélection des files d'attente de transmission définies manuellement par les canaux émetteurs de cluster

Une file d'attente définie manuellement a l'attribut de file d'attente de transmission **USAGE** défini sur XMITQ et l'attribut de nom de canal de cluster **CLCHNAME** défini sur un nom de canal spécifique ou générique.

Si le nom de l'attribut de file d'attente **CLCHNAME** correspond à un nom de canal émetteur de cluster, le canal est associé à la file d'attente. Le nom est soit une correspondance exacte, si le nom ne contient pas de caractères génériques, soit la meilleure correspondance, si le nom contient des caractères génériques.

Si les définitions **CLCHNAME** de plusieurs files d'attente de transmission correspondent au même canal émetteur de cluster, les définitions se chevauchent. Pour résoudre l'ambiguïté, il existe un ordre de priorité entre les correspondances. Les correspondances exactes sont toujours prioritaires. La [Figure 37](#), à la [page 324](#) présente les associations entre les files d'attente de transmission et les canaux émetteurs de cluster. Les flèches noires indiquent les associations réelles et les flèches grises, les associations potentielles. L'ordre de priorité des files d'attente de transmission dans [Figure 37](#), à la [page 324](#) est le suivant:

XMITQ.CL1.QM1

L'attribut **CLCHNAME** de la file d'attente de transmission XMITQ.CL1.QM1 est défini sur CL1.QM1. La définition de l'attribut **CLCHNAME**, CL1.QM1, ne comporte pas de caractères génériques et est prioritaire sur les autres attributs **CLCHNAME**, définis dans d'autres files d'attente de transmission, qui correspondent à des caractères génériques. Le gestionnaire de files d'attente stocke tout message de cluster qui doit être transféré par le canal émetteur de cluster CL1.QM1 dans la file d'attente de transmission XMITQ.CL1.QM1. La seule exception est si l'attribut **CLCHNAME** de plusieurs files d'attente de transmission est défini sur CL1.QM1. Dans ce cas, le gestionnaire de files d'attente stocke les messages du canal émetteur de cluster CL1.QM1 dans l'une de ces files d'attente. Il sélectionne une file d'attente de manière arbitraire lorsque le canal démarre. Il peut sélectionner une autre file d'attente lorsque le canal redémarre.

XMITQ.CL1

L'attribut **CLCHNAME** de la file d'attente de transmission XMITQ.CL1 est défini sur CL1.*. La définition de l'attribut **CLCHNAME**, CL1.*, comporte un caractère générique de fin, qui correspond au nom de tout canal émetteur de cluster commençant par CL1.. Le gestionnaire de files d'attente stocke tous les messages de cluster qui doivent être transférés par un canal émetteur de cluster dont le nom commence par CL1. dans la file d'attente de transmission XMITQ.CL1, sauf s'il existe une file d'attente de transmission avec une correspondance plus spécifique, telle que la file d'attente XMITQ.CL1.QM1. Un caractère générique de fin rend la définition moins spécifique qu'une définition sans caractère générique, et plus spécifique qu'une définition avec plusieurs caractères génériques, ou des caractères génériques qui sont suivis de caractères de fin supplémentaires.

XMITQ.CL.QM

XMITQ.CL.QM est le nom de la file d'attente de transmission avec son attribut **CLCHNAME** défini sur CL*.QM*. La définition de CL*.QM* comporte deux caractères génériques qui correspondent au

nom de tout canal émetteur de cluster qui commence par CL . et qui inclut ou se termine par QM. La correspondance est moins spécifique qu'une correspondance avec un caractère générique.

SYSTEM.CLUSTER.TRANSMIT. *channelName* | QUEUE

Si aucune file d'attente de transmission ne possède d'attribut **CLCHNAME** qui correspond au nom du canal émetteur de cluster que le gestionnaire de files d'attente doit utiliser, le gestionnaire de files d'attente utilise la file d'attente de transmission de cluster par défaut. La file d'attente de transmission de cluster par défaut est soit la file d'attente de transmission de cluster système unique, SYSTEM.CLUSTER.TRANSMIT.QUEUE, soit une file d'attente de transmission de cluster système que le gestionnaire de files d'attente a créée pour un canal émetteur de cluster spécifique, SYSTEM.CLUSTER.TRANSMIT. *channelName*. La file d'attente par défaut dépend de la valeur de l'attribut **DEFXMITQ** du gestionnaire de files d'attente.

Conseil : A moins que vous n'ayez clairement besoin de définitions qui se chevauchent, évitez qu'elles ne conduisent à des configurations compliquées difficiles à comprendre.

Fonctionnement du processus de basculement du canal émetteur de cluster vers une autre file d'attente de transmission

Pour modifier l'association des canaux émetteurs de cluster aux files d'attente de transmission de cluster, modifiez à tout moment le paramètre **CLCHNAME** de toute file d'attente de transmission ou le paramètre de gestionnaire de files d'attente **DEFCLXQ**. Rien ne se passe immédiatement. Les modifications ne se produisent que lorsqu'un canal démarre. Lorsqu'il démarre, il vérifie s'il faut continuer à transférer des messages à partir de la même file d'attente de transmission. Trois types de modification modifient l'association d'un canal émetteur de cluster à une file d'attente de transmission.

1. Redéfinition du paramètre **CLCHNAME** de la file d'attente de transmission à laquelle le canal émetteur de cluster est actuellement associé pour être moins spécifique ou vide, ou suppression de la file d'attente de transmission de cluster lorsque le canal est arrêté.

Certaines autres files d'attente de transmission de cluster peuvent désormais correspondre mieux au nom de canal. Ou, si aucune autre file d'attente de transmission ne correspond au nom du canal émetteur de cluster, l'association doit revenir à la file d'attente de transmission par défaut.

2. Redéfinition du paramètre **CLCHNAME** de toute autre file d'attente de transmission de cluster ou ajout d'une file d'attente de transmission de cluster.

Le paramètre **CLCHNAME** d'une autre file d'attente de transmission peut désormais correspondre mieux au canal émetteur de cluster que la file d'attente de transmission à laquelle le canal émetteur de cluster est actuellement associé. Si le canal émetteur de cluster est actuellement associé à une file d'attente de transmission de cluster par défaut, il peut être associé à une file d'attente de transmission de cluster définie manuellement.

3. Si le canal émetteur de cluster est actuellement associé à une file d'attente de transmission de cluster par défaut, modifiez le paramètre de gestionnaire de files d'attente **DEFCLXQ**.

Si l'association d'un canal émetteur de cluster change, lorsque le canal démarre, il bascule son association vers la nouvelle file d'attente de transmission. Lors du basculement, il garantit qu'aucun message n'est perdu. Les messages sont transférés dans la nouvelle file d'attente de transmission dans l'ordre dans lequel le canal transfère les messages au gestionnaire de files d'attente éloignées.

A faire : Comme pour tout transfert de messages dans un cluster, vous devez placer les messages dans des groupes pour vous assurer que les messages qui doivent être distribués dans l'ordre sont distribués dans l'ordre. Dans de rares cas, les messages peuvent être dans le désordre dans un cluster.

Le processus de commutation passe par les étapes transactionnelles suivantes. Si le processus de commutation est interrompu, l'étape transactionnelle en cours est reprise lorsque le canal redémarre.

Etape 1-Traiter les messages de la file d'attente de transmission d'origine

Le canal émetteur de cluster est associé à la nouvelle file d'attente de transmission, qu'il peut partager avec d'autres canaux émetteurs de cluster. Les messages du canal émetteur de cluster continuent d'être placés dans la file d'attente de transmission d'origine. Un processus de commutation transitoire transfère les messages de la file d'attente de transmission d'origine vers la

nouvelle file d'attente de transmission. Le canal émetteur de cluster transmet les messages de la nouvelle file d'attente de transmission au canal récepteur de cluster. Le statut du canal indique le canal émetteur de cluster toujours associé à l'ancienne file d'attente de transmission.

Le processus de commutation continue également à transférer les messages nouvellement arrivés. Cette étape se poursuit jusqu'à ce que le nombre de messages restants à transmettre par le processus de commutation atteigne zéro. Lorsque le nombre de messages atteint zéro, la procédure passe à l'étape 2.

Au cours de l'étape 1, l'activité du disque pour le canal augmente. Les messages persistants sont validés hors de la première file d'attente de transmission et dans la deuxième file d'attente de transmission. Cette activité de disque s'ajoute aux messages en cours de validation lorsqu'ils sont placés dans et supprimés de la file d'attente de transmission dans le cadre du transfert normal des messages. Idéalement, aucun message n'arrive pendant le processus de commutation, de sorte que la transition peut avoir lieu le plus rapidement possible. Si des messages arrivent, ils sont traités par le processus de commutation.

Etape 2-Traiter les messages de la nouvelle file d'attente de transmission

Dès qu'aucun message ne reste dans la file d'attente de transmission d'origine pour le canal émetteur de cluster, de nouveaux messages sont placés directement dans la nouvelle file d'attente de transmission. Le statut du canal indique que le canal émetteur de cluster est associé à la nouvelle file d'attente de transmission. Le message suivant est consigné dans le journal des erreurs du gestionnaire de files d'attente: " AMQ7341 La file d'attente de transmission du canal *ChannelName* est *QueueName* ."

Attributs de files d'attente de transmission de cluster et de files d'attente de transmission de cluster multiples

Vous avez la possibilité d'acheminer des messages de cluster vers différents gestionnaires de files d'attente stockant les messages dans une ou plusieurs files d'attente de transmission de cluster. Avec une file d'attente, vous disposez d'un ensemble d'attributs de file d'attente de transmission de cluster à définir et à interroger ; avec plusieurs files d'attente, vous disposez de plusieurs ensembles. Pour certains attributs, le fait d'avoir plusieurs ensembles est un avantage: par exemple, l'interrogation de la longueur de la file d'attente vous indique combien de messages sont en attente d'être réacheminés par un ou un ensemble de canaux, plutôt que par tous les canaux. Pour les autres attributs, avoir plusieurs ensembles est un inconvénient: par exemple, vous ne souhaitez probablement pas configurer les mêmes droits d'accès pour chaque file d'attente de transmission de cluster. Pour cette raison, les droits d'accès sont toujours vérifiés par rapport au profil de SYSTEM.CLUSTER.TRANSMIT.QUEUE et non par rapport aux profils d'une file d'attente de transmission de cluster particulière. Si vous souhaitez appliquer des contrôles de sécurité plus granulaires, voir [Contrôle d'accès et plusieurs files d'attente de transmission de cluster](#).

Plusieurs canaux émetteurs de cluster et plusieurs files d'attente de transmission

Un gestionnaire de files d'attente stocke un message dans une file d'attente de transmission de cluster avant de l'acheminer sur un canal émetteur de cluster. Il sélectionne un canal émetteur de cluster qui est connecté à la destination du message. Il peut avoir un choix de canaux émetteurs de cluster qui se connectent tous à la même destination. La destination peut être la même file d'attente physique, connectée par plusieurs canaux émetteurs de cluster à un seul gestionnaire de files d'attente. La destination peut également être un grand nombre de files d'attente physiques avec le même nom de file d'attente, hébergées sur des gestionnaires de files d'attente différents dans le même cluster. Lorsqu'il existe un choix de canaux émetteurs de cluster connectés à une destination, l'algorithme d'équilibrage de charge en choisit un. Le choix dépend d'un certain nombre de facteurs ; voir [Algorithme de gestion de la charge de travail du cluster](#).

Dans [Figure 38](#), à la [page 327](#), CL1.QM1, CL1.QM2 et CS.QM1 sont tous les canaux qui peuvent conduire à la même destination. Par exemple, si vous définissez Q1 dans CL1 sur QM1 et QM2, CL1.QM1 et CL1.QM2 fournissent tous deux des routes vers la même destination, Q1, sur deux gestionnaires de files d'attente différents. Si le canal CS.QM1 se trouve également dans CL1, il s'agit également d'un canal qu'un message pour Q1 peut prendre. L'appartenance à un cluster de CS.QM1 peut être définie par

une liste de noms de cluster, ce qui explique pourquoi le nom de canal n'inclut pas de nom de cluster dans sa construction. En fonction des paramètres d'équilibrage de charge et de l'application émettrice, certains messages pour Q1 peuvent être placés dans chacune des files d'attente de transmission, XMITQ.CL1.QM1, XMITQ.CL1 et SYSTEM.CLUSTER.TRANSMIT.CS.QM1.

Si vous prévoyez de séparer le trafic des messages, de sorte que les messages d'une même destination ne partagent pas de files d'attente ou de canaux avec des messages de destinations différentes, vous devez d'abord envisager de diviser le trafic sur différents canaux émetteurs de cluster, puis de séparer les messages d'un canal particulier sur une file d'attente de transmission différente. Les files d'attente de cluster sur le même cluster, sur le même gestionnaire de files d'attente, partagent normalement les mêmes canaux de cluster. La définition de plusieurs files d'attente de transmission de cluster seule n'est pas suffisante pour séparer le trafic des messages de cluster sur des files d'attente différentes. Sauf si vous séparez les messages des différentes files d'attente de destination sur différents canaux, les messages partagent la même file d'attente de transmission de cluster.

Un moyen simple de séparer les canaux pris par les messages consiste à créer plusieurs clusters. Sur n'importe quel gestionnaire de files d'attente de chaque cluster, définissez une seule file d'attente de cluster. Ensuite, si vous définissez un canal récepteur de cluster différent pour chaque combinaison cluster / gestionnaire de files d'attente, les messages de chaque file d'attente de cluster ne partagent pas de canal de cluster avec les messages des autres files d'attente de cluster. Si vous définissez des files d'attente de transmission distinctes pour les canaux de cluster, le gestionnaire de files d'attente d'envoi stocke les messages pour une seule file d'attente de cluster dans chaque file d'attente de transmission. Par exemple, si vous souhaitez que deux files d'attente de cluster ne partagent pas de ressources, vous pouvez les placer dans des clusters différents sur le même gestionnaire de files d'attente ou sur des gestionnaires de files d'attente différents dans le même cluster.

Le choix de la file d'attente de transmission du cluster n'affecte pas l'algorithme d'équilibrage de charge. L'algorithme d'équilibrage de la charge de travail choisit le canal émetteur de cluster à utiliser pour transmettre un message. Il place le message dans la file d'attente de transmission qui est traitée par ce canal. Si l'algorithme d'équilibrage de charge est appelé pour effectuer une nouvelle sélection, par exemple si le canal s'arrête, il peut être en mesure de sélectionner un autre canal pour transmettre le message. S'il choisit un autre canal et que le nouveau canal achemine les messages à partir d'une file d'attente de transmission de cluster différente, l'algorithme d'équilibrage de la charge de travail transfère le message à l'autre file d'attente de transmission.

Dans [Figure 38](#), à la [page 327](#), deux canaux émetteurs de cluster, CS.QM1 et CS.QM2, sont associés à la file d'attente de transmission du système par défaut. Lorsque l'algorithme d'équilibrage de charge stocke un message dans SYSTEM.CLUSTER.TRANSMIT.QUEUE ou dans toute autre file d'attente de transmission de cluster, le nom du canal émetteur de cluster qui doit transmettre le message est stocké dans l'ID de corrélation du message. Chaque canal transmet uniquement les messages qui correspondent à l'ID de corrélation avec le nom du canal.

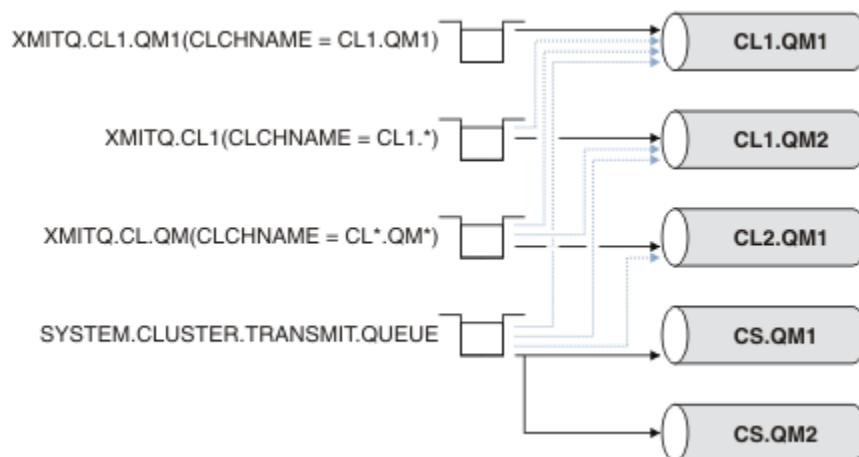


Figure 38. Plusieurs canaux émetteurs de cluster

Si CS.QM1 s'arrête, les messages de la file d'attente de transmission pour ce canal émetteur de cluster sont examinés. Les messages qui peuvent être réacheminés par un autre canal sont traités à nouveau par l'algorithme d'équilibrage de charge. Leur ID de corrélation est réinitialisé sur un autre nom de canal émetteur de cluster. Si l'autre canal émetteur de cluster est CS.QM2, le message est conservé sur SYSTEM.CLUSTER.TRANSMIT.QUEUE. Si l'autre canal est CL1.QM1, l'algorithme d'équilibrage de charge transfère le message à XMITQ.CL1.QM1. Lorsque le canal émetteur de cluster redémarre, les nouveaux messages et les messages qui n'ont pas été marqués pour un autre canal émetteur de cluster sont transférés à nouveau par le canal.

Vous pouvez modifier l'association entre les files d'attente de transmission et les canaux émetteurs de cluster sur un système en cours d'exécution. Vous pouvez modifier un paramètre **CLCHNAME** dans une file d'attente de transmission ou modifier le paramètre de gestionnaire de files d'attente **DEFCLXQ**. Lorsqu'un canal affecté par la modification redémarre, il démarre le processus de commutation de la file d'attente de transmission ; voir [«Fonctionnement du processus de basculement du canal émetteur de cluster vers une autre file d'attente de transmission»](#), à la page 325.

Le processus de basculement de la file d'attente de transmission démarre lorsque le canal est redémarré. Le processus de rééquilibrage de la charge de travail démarre lorsque le canal est arrêté. Les deux processus peuvent s'exécuter en parallèle.

Le cas simple est que l'arrêt d'un canal émetteur de cluster n'entraîne pas le processus de rééquilibrage pour modifier le canal émetteur de cluster qui doit réacheminer les messages de la file d'attente. Dans ce cas, aucun autre canal émetteur de cluster ne peut transmettre les messages à la destination appropriée. Si aucun autre canal émetteur de cluster n'est utilisé pour transmettre les messages à leur destination, les messages restent marqués pour le même canal émetteur de cluster après l'arrêt du canal émetteur de cluster. Lorsque le canal démarre, si un commutateur est en attente, les processus de commutation déplacent les messages vers une autre file d'attente de transmission où ils sont traités par le même canal émetteur de cluster.

Le cas le plus complexe est celui où plusieurs canaux émetteurs de cluster peuvent traiter certains messages vers la même destination. Vous arrêtez et redémarrez le canal émetteur de cluster pour déclencher le commutateur de file d'attente de transmission. Dans de nombreux cas, lorsque vous redémarrez le canal, l'algorithme d'équilibrage de la charge de travail a déjà déplacé des messages de la file d'attente de transmission d'origine vers des files d'attente de transmission différentes servies par des canaux émetteurs de cluster différents. Seuls les messages qui ne peuvent pas être transférés par un autre canal émetteur de cluster restent à transférer dans la nouvelle file d'attente de transmission. Dans certains cas, si le canal est redémarré rapidement, certains messages qui pourraient être transférés par l'algorithme d'équilibrage de charge restent. Dans ce cas, certains messages restants sont commutés par le processus d'équilibrage de la charge de travail et d'autres par le processus de commutation de la file d'attente de transmission.

Concepts associés

[Canaux de cluster](#)

[Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster](#)

[«Calcul de la taille du journal», à la page 710](#)

[Estimation de la taille du journal dont un gestionnaire de files d'attente a besoin.](#)

Tâches associées

[Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster](#)

[«Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle», à la page 367](#)

Suivez les instructions de la tâche pour construire des clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle. Utilisez les clusters comme point de départ pour les exemples suivants qui permettent d'isoler des messages d'une application à partir de messages vers d'autres applications d'un cluster.

[«Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345](#)

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

«Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 376

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une file d'attente de transmission de cluster supplémentaire pour séparer le trafic de messages vers un seul gestionnaire de files d'attente dans un cluster.

«Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 379


Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. , la solution utilise un cluster supplémentaire pour isoler les messages dans une file d'attente de cluster particulière.

Etablissement de la communication dans un cluster

Un initiateur de canal est nécessaire pour démarrer un canal de communication lorsqu'il y a un message à distribuer. Un programme d'écoute de canal attend de démarrer l'autre extrémité d'un canal pour recevoir le message.

Avant de commencer

Pour établir la communication entre les gestionnaires de files d'attente d'un cluster, configurez une liaison à l'aide de l'un des protocoles de communication pris en charge. Les protocoles pris en charge sont les suivants:

- TCP ou LU 6.2 sur n'importe quelle plateforme
-  NetBIOS ou SPX sur les systèmes Windows


Dans le cadre de cette configuration, vous avez également besoin d'initiateurs de canal et de programmes d'écoute de canal, comme vous le faites avec la mise en file d'attente répartie.


Pourquoi et quand exécuter cette tâche

Tous les gestionnaires de files d'attente de cluster ont besoin d'un initiateur de canal pour surveiller la file d'attente d'initialisation définie par le système SYSTEM.CHANNEL.INITQ. SYSTEM.CHANNEL.INITQ est la file d'attente d'initialisation de toutes les files d'attente de transmission, y compris la file d'attente de transmission du cluster.

Chaque gestionnaire de files d'attente doit disposer d'un programme d'écoute de canal. Un programme d'écoute de canal attend les demandes réseau entrantes et démarre le canal récepteur approprié lorsqu'il est nécessaire. L'implémentation des programmes d'écoute de canal est spécifique à la plateforme, mais il existe des fonctionnalités communes.

Sur toutes les plateformes IBM MQ, le programme d'écoute peut être démarré à l'aide de la commande **START LISTENER**.

 Sur Multiplatforms, vous pouvez démarrer le programme d'écoute automatiquement en même temps que le gestionnaire de files d'attente. Pour démarrer le programme d'écoute automatiquement, définissez l'attribut CONTROL de l'objet LISTENER sur QMGR ou STARTONLY.

 Un port d'écoute non partagé (INDISP (QMGR)) doit être utilisé pour les canaux CLUSRCVR sur z/OS et pour les canaux CLUSSDR vers z/OS.

Procédure

1. Lancez l'initiateur de canal.

- ▶ **z/OS** Sous z/OS, il existe un initiateur de canal pour chaque gestionnaire de files d'attente et il s'exécute en tant qu'espace adresse distinct. Vous le démarrez à l'aide de la commande **MQSC START CHINIT**, que vous émettez dans le cadre du démarrage de votre gestionnaire de files d'attente.
- ▶ **ALW** Sous AIX, Linux, and Windows, lorsque vous démarrez un gestionnaire de files d'attente, si l'attribut de gestionnaire de files d'attente SCHINIT est défini sur QMGR, un initiateur de canal est automatiquement démarré. Sinon, il peut être démarré à l'aide de la commande **runmqsc START CHINIT** ou de la commande de contrôle **runmqchi**.
- ▶ **IBM i** Sous IBM i, lorsque vous démarrez un gestionnaire de files d'attente, si l'attribut de gestionnaire de files d'attente SCHINIT est défini sur QMGR, un initiateur de canal est automatiquement démarré. Sinon, il peut être démarré à l'aide de la commande **runmqsc START CHINIT** ou de la commande de contrôle **runmqchi**.

2. Démarrez le programme d'écoute de canal.

- ▶ **z/OS** Sous z/OS, utilisez le programme d'écoute de canal fourni par IBM MQ. Pour démarrer un programme d'écoute de canal IBM MQ, utilisez la commande **MQSC START LISTENER**, que vous émettez dans le cadre du démarrage de l'initiateur de canal. Exemple :

```
START LISTENER PORT(1414) TRPTYPE(TCP)
```

ou:

```
START LISTENER LUNAME(LONDON.LUNAME) TRPTYPE(LU62)
```

Les membres d'un groupe de partage de files d'attente peuvent utiliser un programme d'écoute partagé au lieu d'un programme d'écoute pour chaque gestionnaire de files d'attente. N'utilisez pas de programmes d'écoute partagés avec des clusters. En particulier, ne définissez pas CONNAME du canal CLUSRCVR comme adresse du programme d'écoute partagé du groupe de partage de files d'attente. Dans ce cas, les gestionnaires de files d'attente peuvent recevoir des messages pour les files d'attente pour lesquelles ils n'ont pas de définition.

- ▶ **IBM i** Sous IBM i, utilisez le programme d'écoute de canal fourni par IBM MQ. Pour démarrer un programme d'écoute de canal IBM MQ, utilisez la commande **CL STRMQLSR**. Exemple :

```
STRMQLSR MQMNAME(QM1) PORT(1414)
```

- ▶ **Windows** Sous Windows, utilisez le programme d'écoute de canal fourni par IBM MQ ou les fonctions fournies par le système d'exploitation.

Pour démarrer le programme d'écoute de canal IBM MQ, utilisez la commande **RUNMQLSR**. Exemple :

```
RUNMQLSR -t tcp -p 1414 -m QM1
```

- ▶ **Linux** ▶ **AIX** Sous AIX and Linux, utilisez le programme d'écoute de canal fourni par IBM MQ ou les fonctions fournies par le système d'exploitation ; par exemple, **inetd** pour les communications TCP.

Pour démarrer le programme d'écoute de canal IBM MQ , utilisez la commande **runmqclsr** .

Exemple :

```
runmqclsr -t tcp -p 1414 -m QM1
```

Pour utiliser **inetd** afin de démarrer des canaux, configurez deux fichiers:

- a. Editez le fichier `/etc/services`. Vous devez être connecté en tant que superutilisateur ou superutilisateur. Si la ligne suivante ne figure pas dans le fichier, ajoutez-la comme indiqué:

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

où 1414 est le numéro de port requis par IBM MQ. Vous pouvez modifier le numéro de port, mais il doit correspondre au numéro de port spécifié à l'extrémité émettrice.

- b. Editez le fichier `/etc/inetd.conf`. Si vous ne disposez pas de la ligne suivante dans ce fichier, ajoutez-la comme suit:

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

où `MQ_INSTALLATION_PATH` est remplacé par le répertoire de haut niveau dans lequel IBM MQ est installé.

Les mises à jour deviennent actives une fois que **inetd** a relu les fichiers de configuration. Exécutez les commandes suivantes à partir de l'ID utilisateur root:

AIX

Sous AIX :

```
refresh -s inetd
```

Linux

Sous Linux:

- a. Recherchez l'ID de processus de **inetd** à l'aide de la commande suivante:

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

- b. Exécutez la commande appropriée.

Pour Linux:

```
kill -1 inetd processid
```

Configuration d'un nouveau cluster

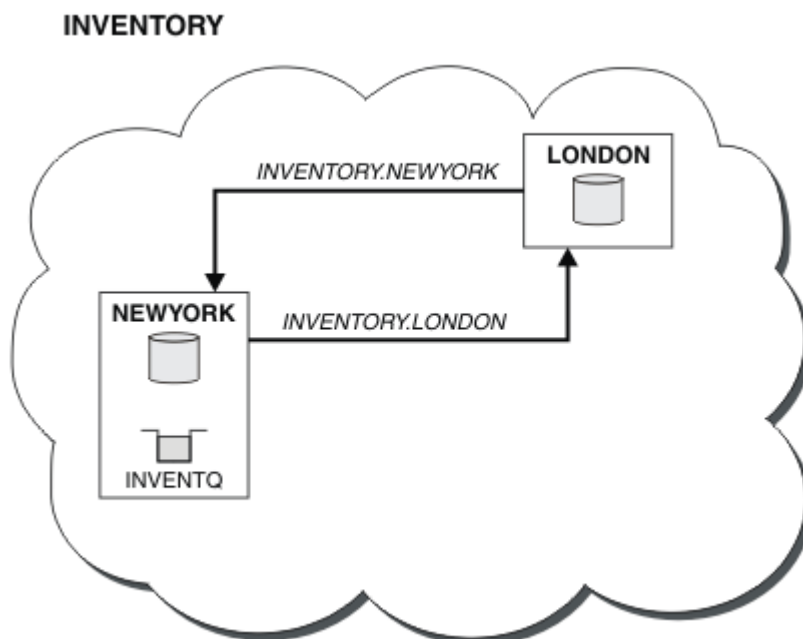
Suivez ces instructions pour configurer l'exemple de cluster. Des instructions distinctes décrivent la configuration du cluster sur TCP/IP, LU 6.2 et avec une ou plusieurs files d'attente de transmission. Testez le cluster en envoyant un message d'un gestionnaire de files d'attente à l'autre.

Avant de commencer

- Au lieu de suivre ces instructions, vous pouvez utiliser l'un des assistants fournis avec IBM MQ Explorer pour créer un cluster comme celui créé par cette tâche. Cliquez avec le bouton droit de la souris sur le dossier Clusters de gestionnaires de files d'attente, puis cliquez sur **Nouveau > Cluster de gestionnaires de files d'attente** et suivez les instructions de l'assistant.
- Pour plus d'informations sur la procédure de configuration d'un cluster, voir [«Définition de files d'attente de cluster»](#), à la page 317, [Canaux de cluster](#) et [Programmes d'écoute](#).

Pourquoi et quand exécuter cette tâche

Vous configurez un nouveau réseau IBM MQ pour un magasin de chaîne. Le magasin a deux succursales, l'une à Londres et l'autre à New York. Les données et les applications de chaque magasin sont hébergées par des systèmes exécutant des gestionnaires de files d'attente distincts. Les deux gestionnaires de files d'attente sont appelés LONDON et NEWYORK. L'application d'inventaire s'exécute sur le système de New York, connectée au gestionnaire de files d'attente NEWYORK. L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ, hébergée par NEWYORK. Les deux gestionnaires de files d'attente, LONDON et NEWYORK, doivent être liés dans un cluster appelé INVENTORY afin qu'ils puissent tous deux insérer des messages dans INVENTQ.




Voici à quoi ressemble ce cluster:

Vous pouvez configurer chaque gestionnaire de files d'attente du cluster pour envoyer des messages à d'autres gestionnaires de files d'attente du cluster à l'aide de différentes files d'attente de transmission de cluster.

Les instructions de configuration du cluster varient légèrement en fonction du protocole de transport, du nombre de files d'attente de transmission ou de la plateforme. Vous avez le choix entre trois combinaisons. La procédure de vérification reste la même pour toutes les combinaisons.

INVENTORY est un petit cluster. Cependant, il est utile comme preuve de concept. Ce qu'il est important de comprendre à propos de ce cluster, c'est la portée qu'il offre pour les améliorations futures.

Procédure

- [«Configuration d'un cluster à l'aide de TCP/IP avec une seule file d'attente de transmission par gestionnaire de files d'attente», à la page 333](#)
- [«Configuration d'un cluster sur TCP/IP à l'aide de plusieurs files d'attente de transmission par gestionnaire de files d'attente», à la page 336](#)
-  [«Configuration d'un cluster à l'aide de LU 6.2 sous z/OS», à la page 339](#)
- [«Vérification du cluster», à la page 341](#)

Concepts associés

[Clusters](#)

[Comparaison de la mise en cluster et de la mise en file d'attente répartie](#)

[Composants d'un cluster](#)

Tâches associées

«Configuration d'un cluster de gestionnaires de files d'attente», à la page 316

Les clusters fournissent un mécanisme d'interconnexion des gestionnaires de files d'attente qui simplifie à la fois la configuration initiale et la gestion en cours. Vous pouvez définir des composants de cluster et créer et gérer des clusters.

Configuration d'un cluster à l'aide de TCP/IP avec une seule file d'attente de transmission par gestionnaire de files d'attente


Il s'agit de l'une des trois rubriques décrivant les différentes configurations d'un cluster simple.

Avant de commencer

Pour une présentation du cluster en cours de création, voir «Configuration d'un nouveau cluster», à la page 331.

L'attribut de gestionnaire de files d'attente, **DEFCLXQ**, doit être laissé comme valeur par défaut, SCTQ.

Pourquoi et quand exécuter cette tâche

Suivez ces étapes pour configurer un cluster sur [Multiplateformes](#) à l'aide du protocole de transport TCP/IP.  Sous z/OS, vous devez suivre les instructions de la rubrique «Defining a TCP connection on z/OS», à la page 1030 pour configurer la connexion TCP/IP au lieu de définir les programmes d'écoute à l'étape «4», à la page 334. Sinon, les étapes sont les mêmes pour z/OS, mais les messages d'erreur sont écrits sur la console et non dans le journal des erreurs du gestionnaire de files d'attente.

Procédure

1. Décidez de l'organisation du cluster et de son nom.

Vous avez décidé de lier les deux gestionnaires de files d'attente, LONDON et NEWYORK, dans un cluster. Un cluster avec seulement deux gestionnaires de files d'attente offre un avantage marginal sur un réseau qui doit utiliser la mise en file d'attente répartie. C'est une bonne façon de commencer et cela offre des possibilités d'expansion future. Lorsque vous ouvrez de nouvelles branches de votre magasin, vous pouvez facilement ajouter les nouveaux gestionnaires de files d'attente au cluster. L'ajout de nouveaux gestionnaires de files d'attente ne perturbe pas le réseau existant ; voir «Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343.

Pour le moment, la seule application que vous exécutez est l'application d'inventaire. Le nom du cluster est INVENTORY.

2. Choisissez les gestionnaires de files d'attente qui doivent contenir les référentiels complets.

Dans n'importe quel cluster, vous devez désigner au moins un gestionnaire de files d'attente, ou de préférence deux, pour conserver les référentiels complets. Dans cet exemple, il n'y a que deux gestionnaires de files d'attente, LONDON et NEWYORK, qui contiennent tous deux des référentiels complets.

- a. Vous pouvez effectuer les étapes restantes dans n'importe quel ordre.
- b. Au fur et à mesure que vous suivez les étapes, des messages d'avertissement peuvent être écrits dans le journal du gestionnaire de files d'attente. Les messages sont le résultat de définitions manquantes que vous n'avez pas encore ajoutées.

Examples of the responses to the commands are shown in a box like this after each step in this task. These examples show the responses returned by IBM MQ for AIX. The responses vary on other platforms.

- c. Avant de procéder à ces étapes, assurez-vous que les gestionnaires de files d'attente sont démarrés.
3. Modifiez les définitions de gestionnaire de files d'attente pour ajouter des définitions de référentiel.

Sur chaque gestionnaire de files d'attente devant contenir un référentiel complet, modifiez la définition du gestionnaire de files d'attente local à l'aide de la commande ALTER QMGR et en spécifiant l'attribut REPOS :

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: IBM MQ queue manager changed.
```

Par exemple, si vous entrez:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON est remplacé par un référentiel complet.

4. Définissez les programmes d'écoute.

Définissez un programme d'écoute qui accepte les demandes réseau provenant d'autres gestionnaires de files d'attente pour chaque gestionnaire de files d'attente du cluster. Sur les gestionnaires de files d'attente LONDON , exécutez la commande suivante:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

L'attribut CONTROL garantit que le programme d'écoute démarre et s'arrête lorsque le gestionnaire de files d'attente le fait.

Le programme d'écoute n'étant pas démarré lorsqu'il est défini, il doit être démarré manuellement la première fois à l'aide de la commande MQSC suivante:

```
START LISTENER(LONDON_LS)
```

Exécutez des commandes similaires pour tous les autres gestionnaires de files d'attente du cluster, en modifiant le nom du programme d'écoute pour chacun d'eux.

Il existe plusieurs façons de définir ces programmes d'écoute, comme indiqué dans [Programmes d'écoute](#).

5. Définissez le canal CLUSRCVR pour le gestionnaire de files d'attente LONDON .


Sur chaque gestionnaire de files d'attente d'un cluster, vous définissez un canal récepteur de cluster sur lequel le gestionnaire de files d'attente peut recevoir des messages. Voir [Canal récepteur de cluster: CLUSRCVR](#) . Le canal CLUSRCVR définit le nom de connexion du gestionnaire de files d'attente. Le nom de connexion est stocké dans les référentiels, où d'autres gestionnaires de files d'attente peuvent y faire référence. Le mot clé CLUSTER indique la disponibilité du gestionnaire de files d'attente pour recevoir des messages d'autres gestionnaires de files d'attente du cluster.

Dans cet exemple, le nom de canal est INVENTORY . LONDON est le nom de connexion (CONNAME) est l'adresse réseau de la machine sur laquelle réside le gestionnaire de files d'attente, à savoir LONDON .CHSTORE .COM. L'adresse réseau peut être entrée sous la forme d'un nom d'hôte DNS alphanumérique ou d'une adresse IP au format décimal à points IPv4 . Par exemple, 192 .0 .2 .0 ou IPv6 format hexadécimal, par exemple 2001 :DB8 :0204 :acff :fe97 :2c34 :fde0 :3485. Le numéro de port n'étant pas spécifié, le port par défaut (1414) est utilisé.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. Définissez le canal CLUSRCVR pour le gestionnaire de files d'attente NEWYORK .

 Si le programme d'écoute de canal utilise le port par défaut, généralement 1414, et que le cluster n'inclut pas de gestionnaire de files d'attente sous z/OS, vous pouvez omettre CONNAME.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
```

7. Définissez le canal CLUSSDR sur le gestionnaire de files d'attente LONDON .

Vous définissez manuellement un canal CLUSSDR entre chaque gestionnaire de files d'attente de référentiel complet et tout autre gestionnaire de files d'attente de référentiel complet dans le cluster. Voir [Canal émetteur de cluster: CLUSSDR](#) . Dans ce cas, il n'y a que deux gestionnaires de files d'attente, qui contiennent tous deux des référentiels complets. Ils ont chacun besoin d'un canal CLUSSDR défini manuellement qui pointe vers le canal CLUSRCVR défini sur l'autre gestionnaire de files d'attente. Les noms de canal indiqués dans les définitions CLUSSDR doivent correspondre aux noms de canal indiqués dans les définitions CLUSRCVR correspondantes. Lorsqu'un gestionnaire de files d'attente possède des définitions pour un canal récepteur de cluster et un canal émetteur de cluster dans le même cluster, le canal émetteur de cluster est démarré.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: WebSphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

8. Définissez le canal CLUSSDR sur le gestionnaire de files d'attente NEWYORK .

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. Définition de la file d'attente de cluster INVENTQ

Définissez la file d'attente INVENTQ sur le gestionnaire de files d'attente NEWYORK en spécifiant le mot clé CLUSTER .

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: WebSphere MQ queue created.
```

Le mot clé CLUSTER permet d'annoncer la file d'attente au cluster. Dès que la file d'attente est définie, elle devient disponible pour les autres gestionnaires de files d'attente du cluster. Ils peuvent lui envoyer des messages sans avoir à créer une définition de file d'attente éloignée.

Toutes les définitions sont complètes. Sur toutes les plateformes, démarrez un programme d'écoute sur chaque gestionnaire de files d'attente. Le programme d'écoute attend les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Que faire ensuite

Vous êtes maintenant prêt à [vérifier le cluster](#).

Tâches associées

«[Configuration d'un cluster sur TCP/IP à l'aide de plusieurs files d'attente de transmission par gestionnaire de files d'attente](#)», à la page 336

Il s'agit de l'une des trois rubriques décrivant les différentes configurations d'un cluster simple.

«[Configuration d'un cluster à l'aide de LU 6.2 sous z/OS](#)», à la page 339

Il s'agit de l'une des rubriques d'arborescence décrivant différentes configurations pour un cluster simple.

Configuration d'un cluster sur TCP/IP à l'aide de plusieurs files d'attente de transmission par gestionnaire de files d'attente

Il s'agit de l'une des trois rubriques décrivant les différentes configurations d'un cluster simple.

Avant de commencer

Pour une présentation du cluster en cours de création, voir «[Configuration d'un nouveau cluster](#)», à la page 331.

Pourquoi et quand exécuter cette tâche

Suivez ces étapes pour configurer un cluster sur [Multiplateformes](#) à l'aide du protocole de transport TCP/IP. Les gestionnaires de files d'attente de référentiel sont configurés pour utiliser une file d'attente de transmission de cluster différente pour s'envoyer des messages les uns aux autres et aux autres gestionnaires de files d'attente du cluster. Si vous ajoutez des gestionnaires de files d'attente au cluster qui doivent également utiliser des files d'attente de transmission différentes, suivez la tâche «[Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes](#)», à la page 345.

Procédure

1. Décidez de l'organisation du cluster et de son nom.

Vous avez décidé de lier les deux gestionnaires de files d'attente, LONDON et NEWYORK, dans un cluster. Un cluster avec seulement deux gestionnaires de files d'attente offre un avantage marginal sur un réseau qui doit utiliser la mise en file d'attente répartie. C'est une bonne façon de commencer et cela offre des possibilités d'expansion future. Lorsque vous ouvrez de nouvelles branches de votre magasin, vous pouvez facilement ajouter les nouveaux gestionnaires de files d'attente au cluster. L'ajout de nouveaux gestionnaires de files d'attente ne perturbe pas le réseau existant ; voir «[Ajout d'un gestionnaire de files d'attente à un cluster](#)», à la page 343.

Pour le moment, la seule application que vous exécutez est l'application d'inventaire. Le nom du cluster est INVENTORY.

2. Choisissez les gestionnaires de files d'attente qui doivent contenir les référentiels complets.

Dans n'importe quel cluster, vous devez désigner au moins un gestionnaire de files d'attente, ou de préférence deux, pour conserver les référentiels complets. Dans cet exemple, il n'y a que deux gestionnaires de files d'attente, LONDON et NEWYORK, qui contiennent tous deux des référentiels complets.

- a. Vous pouvez effectuer les étapes restantes dans n'importe quel ordre.
- b. Au fur et à mesure que vous suivez les étapes, des messages d'avertissement peuvent être écrits dans le journal du gestionnaire de files d'attente. Les messages sont le résultat de définitions manquantes que vous n'avez pas encore ajoutées.

Examples of the responses to the commands are shown in a box like this after each step in this task. These examples show the responses returned by IBM MQ for AIX. The responses vary on other platforms.

- c. Avant de procéder à ces étapes, assurez-vous que les gestionnaires de files d'attente sont démarrés.
3. Modifiez les définitions de gestionnaire de files d'attente pour ajouter des définitions de référentiel.

Sur chaque gestionnaire de files d'attente devant contenir un référentiel complet, modifiez la définition du gestionnaire de files d'attente local à l'aide de la commande ALTER QMGR et en spécifiant l'attribut REPOS :

```
ALTER QMGR REPOS(INVENTORY)
```

```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: IBM MQ queue manager changed.
```

Par exemple, si vous entrez:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON est remplacé par un référentiel complet.

4. Modifiez les définitions de gestionnaire de files d'attente pour créer des files d'attente de transmission de cluster distinctes pour chaque destination.

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Sur chaque gestionnaire de files d'attente que vous ajoutez au cluster, choisissez d'utiliser ou non des files d'attente de transmission distinctes. Voir les rubriques «Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343 et «Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345.

5. Définissez les programmes d'écoute.

Définissez un programme d'écoute qui accepte les demandes réseau provenant d'autres gestionnaires de files d'attente pour chaque gestionnaire de files d'attente du cluster. Sur les gestionnaires de files d'attente LONDON , exécutez la commande suivante:

```
DEFINE LISTENER(LONDON_LS) TRPTYPE(TCP) CONTROL(QMGR)
```

L'attribut CONTROL garantit que le programme d'écoute démarre et s'arrête lorsque le gestionnaire de files d'attente le fait.

Le programme d'écoute n'étant pas démarré lorsqu'il est défini, il doit être démarré manuellement la première fois à l'aide de la commande MQSC suivante:

```
START LISTENER(LONDON_LS)
```

Exécutez des commandes similaires pour tous les autres gestionnaires de files d'attente du cluster, en modifiant le nom du programme d'écoute pour chacun d'eux.

Il existe plusieurs façons de définir ces programmes d'écoute, comme indiqué dans [Programmes d'écoute](#).

6. Définissez le canal CLUSRCVR pour le gestionnaire de files d'attente LONDON .


Sur chaque gestionnaire de files d'attente d'un cluster, vous définissez un canal récepteur de cluster sur lequel le gestionnaire de files d'attente peut recevoir des messages. Voir [Canal récepteur de cluster: CLUSRCVR](#) . Le canal CLUSRCVR définit le nom de connexion du gestionnaire de files d'attente. Le nom de connexion est stocké dans les référentiels, où d'autres gestionnaires de files d'attente peuvent y faire référence. Le mot clé CLUSTER indique la disponibilité du gestionnaire de files d'attente pour recevoir des messages d'autres gestionnaires de files d'attente du cluster.

Dans cet exemple, le nom de canal est INVENTORY.LONDON et le nom de connexion (CONNAME) est l'adresse réseau de la machine sur laquelle réside le gestionnaire de files d'attente, à savoir LONDON.CHSTORE.COM. L'adresse réseau peut être entrée sous la forme d'un nom d'hôte DNS alphanumérique ou d'une adresse IP au format décimal à points IPv4. Par exemple, 192.0.2.0 ou IPv6 format hexadécimal, par exemple 2001:DB8:0204:acff:fe97:2c34:fde0:3485. Le numéro de port n'étant pas spécifié, le port par défaut (1414) est utilisé.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

7. Définissez le canal CLUSRCVR pour le gestionnaire de files d'attente NEWYORK.

 Si le programme d'écoute de canal utilise le port par défaut, généralement 1414, et que le cluster n'inclut pas de gestionnaire de files d'attente sous z/OS, vous pouvez omettre CONNAME.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('TCP Cluster-receiver channel for queue manager NEWYORK')
```

8. Définissez le canal CLUSSDR sur le gestionnaire de files d'attente LONDON.

Vous définissez manuellement un canal CLUSSDR entre chaque gestionnaire de files d'attente de référentiel complet et tout autre gestionnaire de files d'attente de référentiel complet dans le cluster. Voir [Canal émetteur de cluster: CLUSSDR](#). Dans ce cas, il n'y a que deux gestionnaires de files d'attente, qui contiennent tous deux des référentiels complets. Ils ont chacun besoin d'un canal CLUSSDR défini manuellement qui pointe vers le canal CLUSRCVR défini sur l'autre gestionnaire de files d'attente. Les noms de canal indiqués dans les définitions CLUSSDR doivent correspondre aux noms de canal indiqués dans les définitions CLUSRCVR correspondantes. Lorsqu'un gestionnaire de files d'attente possède des définitions pour un canal récepteur de cluster et un canal émetteur de cluster dans le même cluster, le canal émetteur de cluster est démarré.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: WebSphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

9. Définissez le canal CLUSSDR sur le gestionnaire de files d'attente NEWYORK.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('TCP Cluster-sender channel from NEWYORK to repository at LONDON')
```

10. Définition de la file d'attente de cluster INVENTQ

Définissez la file d'attente INVENTQ sur le gestionnaire de files d'attente NEWYORK en spécifiant le mot clé CLUSTER.

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: WebSphere MQ queue created.
```

Le mot clé CLUSTER permet d'annoncer la file d'attente au cluster. Dès que la file d'attente est définie, elle devient disponible pour les autres gestionnaires de files d'attente du cluster. Ils peuvent lui envoyer des messages sans avoir à créer une définition de file d'attente éloignée.

Toutes les définitions sont complètes. Sur toutes les plateformes, démarrez un programme d'écoute sur chaque gestionnaire de files d'attente. Le programme d'écoute attend les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Que faire ensuite

Vous êtes maintenant prêt à [vérifier le cluster](#).

Tâches associées

«[Configuration d'un cluster à l'aide de TCP/IP avec une seule file d'attente de transmission par gestionnaire de files d'attente](#)», à la page 333

Il s'agit de l'une des trois rubriques décrivant les différentes configurations d'un cluster simple.

«[Configuration d'un cluster à l'aide de LU 6.2 sous z/OS](#)», à la page 339

Il s'agit de l'une des rubriques d'arborescence décrivant différentes configurations pour un cluster simple.

Configuration d'un cluster à l'aide de LU 6.2 sous z/OS

Il s'agit de l'une des rubriques d'arborescence décrivant différentes configurations pour un cluster simple.

Avant de commencer

Pour une présentation du cluster en cours de création, voir «[Configuration d'un nouveau cluster](#)», à la page 331.

Procédure

1. Décidez de l'organisation du cluster et de son nom.

Vous avez décidé de lier les deux gestionnaires de files d'attente, LONDON et NEWYORK, dans un cluster. Un cluster avec seulement deux gestionnaires de files d'attente offre un avantage marginal sur un réseau qui doit utiliser la mise en file d'attente répartie. C'est une bonne façon de commencer et cela offre des possibilités d'expansion future. Lorsque vous ouvrez de nouvelles branches de votre magasin, vous pouvez facilement ajouter les nouveaux gestionnaires de files d'attente au cluster. L'ajout de nouveaux gestionnaires de files d'attente ne perturbe pas le réseau existant ; voir «[Ajout d'un gestionnaire de files d'attente à un cluster](#)», à la page 343.

Pour le moment, la seule application que vous exécutez est l'application d'inventaire. Le nom du cluster est INVENTORY.

2. Choisissez les gestionnaires de files d'attente qui doivent contenir les référentiels complets.

Dans n'importe quel cluster, vous devez désigner au moins un gestionnaire de files d'attente, ou de préférence deux, pour conserver les référentiels complets. Dans cet exemple, il n'y a que deux gestionnaires de files d'attente, LONDON et NEWYORK, qui contiennent tous deux des référentiels complets.

- a. Vous pouvez effectuer les étapes restantes dans n'importe quel ordre.
- b. Au fur et à mesure que vous exécutez les étapes, des messages d'avertissement peuvent être écrits dans la console système z/OS . Les messages sont le résultat de définitions manquantes que vous n'avez pas encore ajoutées.
- c. Avant de procéder à ces étapes, assurez-vous que les gestionnaires de files d'attente sont démarrés.

3. Modifiez les définitions de gestionnaire de files d'attente pour ajouter des définitions de référentiel.

Sur chaque gestionnaire de files d'attente devant contenir un référentiel complet, modifiez la définition du gestionnaire de files d'attente local à l'aide de la commande ALTER QMGR et en spécifiant l'attribut REPOS :

```
ALTER QMGR REPOS(INVENTORY)
```


```
1 : ALTER QMGR REPOS(INVENTORY)
AMQ8005: IBM MQ queue manager changed.
```

Par exemple, si vous entrez:

- a. runmqsc LONDON
- b. ALTER QMGR REPOS(INVENTORY)

LONDON est remplacé par un référentiel complet.

4. Définissez les programmes d'écoute.

 Voir [L'initiateur de canal sous z/OS](#) et «Receiving on LU 6.2», à la page 1034.

Le programme d'écoute n'étant pas démarré lorsqu'il est défini, il doit être démarré manuellement la première fois à l'aide de la commande MQSC suivante:

```
START LISTENER(LONDON_LS)
```

Exécutez des commandes similaires pour tous les autres gestionnaires de files d'attente du cluster, en modifiant le nom du programme d'écoute pour chacun d'eux.

5. Définissez le canal CLUSRCVR pour le gestionnaire de files d'attente LONDON .

Sur chaque gestionnaire de files d'attente d'un cluster, vous définissez un canal récepteur de cluster sur lequel le gestionnaire de files d'attente peut recevoir des messages. Voir [Canal récepteur de cluster: CLUSRCVR](#) . Le canal CLUSRCVR définit le nom de connexion du gestionnaire de files d'attente. Le nom de connexion est stocké dans les référentiels, où d'autres gestionnaires de files d'attente peuvent y faire référence. Le mot clé CLUSTER indique la disponibilité du gestionnaire de files d'attente pour recevoir des messages d'autres gestionnaires de files d'attente du cluster.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
```

```
1 : DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager LONDON')
AMQ8014: WebSphere MQ channel created.
07/09/98 12:56:35 No repositories for cluster 'INVENTORY'
```

6. Définissez le canal CLUSRCVR pour le gestionnaire de files d'attente NEWYORK .

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) TRPTYPE(LU62)
CONNNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-receiver channel for queue manager NEWYORK')
```

7. Définissez le canal CLUSSDR sur le gestionnaire de files d'attente LONDON .

Vous définissez manuellement un canal CLUSSDR entre chaque gestionnaire de files d'attente de référentiel complet et tout autre gestionnaire de files d'attente de référentiel complet dans le cluster. Voir [Canal émetteur de cluster: CLUSSDR](#) . Dans ce cas, il n'y a que deux gestionnaires de files d'attente, qui contiennent tous deux des référentiels complets. Ils ont chacun besoin d'un canal

CLUSSDR défini manuellement qui pointe vers le canal CLUSRCVR défini sur l'autre gestionnaire de files d'attente. Les noms de canal indiqués dans les définitions CLUSSDR doivent correspondre aux noms de canal indiqués dans les définitions CLUSRCVR correspondantes. Lorsqu'un gestionnaire de files d'attente possède des définitions pour un canal récepteur de cluster et un canal émetteur de cluster dans le même cluster, le canal émetteur de cluster est démarré.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNNAME(CPIC) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
```

```
1 : DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNNAME(NEWYORK.LUNAME) CLUSTER(INVENTORY)
MODENAME('#INTER') TPNAME('MQSERIES')
DESCR('LU62 Cluster-sender channel from LONDON to repository at NEWYORK')
AMQ8014: WebSphere MQ channel created.
07/09/98 13:00:18 Channel program started.
```

8. Définissez le canal CLUSSDR sur le gestionnaire de files d'attente NEWYORK .

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(LU62)
CONNNAME(LONDON.LUNAME) CLUSTER(INVENTORY)
DESCR('LU62 Cluster-sender channel from NEWYORK to repository at LONDON')
```

9. Définition de la file d'attente de cluster INVENTQ

Définissez la file d'attente INVENTQ sur le gestionnaire de files d'attente NEWYORK en spécifiant le mot clé CLUSTER .

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

```
1 : DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
AMQ8006: WebSphere MQ queue created.
```

Le mot clé CLUSTER permet d'annoncer la file d'attente au cluster. Dès que la file d'attente est définie, elle devient disponible pour les autres gestionnaires de files d'attente du cluster. Ils peuvent lui envoyer des messages sans avoir à créer une définition de file d'attente éloignée.

Toutes les définitions sont complètes. Sur toutes les plateformes, démarrez un programme d'écoute sur chaque gestionnaire de files d'attente. Le programme d'écoute attend les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Que faire ensuite

Vous êtes maintenant prêt à vérifier le cluster.

Tâches associées

«[Configuration d'un cluster à l'aide de TCP/IP avec une seule file d'attente de transmission par gestionnaire de files d'attente](#)», à la page 333

Il s'agit de l'une des trois rubriques décrivant les différentes configurations d'un cluster simple.

«[Configuration d'un cluster sur TCP/IP à l'aide de plusieurs files d'attente de transmission par gestionnaire de files d'attente](#)», à la page 336


Il s'agit de l'une des trois rubriques décrivant les différentes configurations d'un cluster simple.

Vérification du cluster

Les rubriques homologues décrivent trois configurations différentes pour un cluster simple. Cette rubrique explique comment vérifier le cluster.

Avant de commencer

Cette rubrique suppose que vous vérifiez un cluster que vous avez créé à l'aide de l'une des tâches suivantes:

- «[Configuration d'un cluster à l'aide de TCP/IP avec une seule file d'attente de transmission par gestionnaire de files d'attente](#)», à la page 333.
- «[Configuration d'un cluster sur TCP/IP à l'aide de plusieurs files d'attente de transmission par gestionnaire de files d'attente](#)», à la page 336.
-  «[Configuration d'un cluster à l'aide de LU 6.2 sous z/OS](#)», à la page 339.

Pour une présentation du cluster qui a été créé, voir «[Configuration d'un nouveau cluster](#)», à la page 331.

Pourquoi et quand exécuter cette tâche

Vous pouvez vérifier le cluster d'une ou de plusieurs des manières suivantes:

1. Exécution de commandes d'administration pour afficher les attributs de cluster et de canal.
2. Exécutez les exemples de programme pour envoyer et recevoir des messages dans une file d'attente de cluster.
3. Ecrivez vos propres programmes pour envoyer un message de demande à une file d'attente de cluster et répondre avec des messages de réponse à une file d'attente de réponses non groupée.

Procédure

Exécutez les commandes DISPLAY **runmqsc** pour vérifier le cluster.

Les réponses que vous voyez doivent être similaires à celles des étapes suivantes.

1. A partir du gestionnaire de files d'attente NEWYORK , exécutez la commande **DISPLAY CLUSQMGR** :

```
dis clusqmgr(*)
```

```
1 : dis clusqmgr(*)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(NEWYORK)      CLUSTER(INVENTORY)
CHANNEL(INVENTORY.NEWYORK)
AMQ8441: Display Cluster Queue Manager details.
CLUSQMGR(LONDON)      CLUSTER(INVENTORY)
CHANNEL(INVENTORY.LONDON)
```

2. A partir du gestionnaire de files d'attente NEWYORK , exécutez la commande **DISPLAY CHANNEL STATUS** :

```
dis chstatus(*)
```

```
1 : dis chstatus(*)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.NEWYORK) XMITQ( )
CONNAME(192.0.2.0)         CURRENT
CHLTYPE(CLUSRCVR)         STATUS(RUNNING)
RQMNAME(LONDON)
AMQ8417: Display Channel Status details.
CHANNEL(INVENTORY.LONDON) XMITQ(SYSTEM.CLUSTER.TRANSMIT.INVENTORY.LONDON)
CONNAME(192.0.2.1)         CURRENT
CHLTYPE(CLUSSDR)          STATUS(RUNNING)
RQMNAME(LONDON)
```

Envoyez des messages entre les deux gestionnaires de files d'attente à l'aide de **amqsput**.

3. Sous LONDON , exécutez la commande **amqsput INVENTQ LONDON**.

Entrez des messages, suivis d'une ligne vide.

4. Sous NEWYORK , exécutez la commande **amqsget INVENTQ NEWYORK**.

Vous voyez maintenant les messages que vous avez entrés sur LONDON. Au bout de 15 secondes, le programme s'arrête.

Envoyez des messages entre les deux gestionnaires de files d'attente à l'aide de vos propres programmes.

Dans les étapes suivantes, LONDON insère un message dans INVENTQ à l'adresse NEWYORK et reçoit une réponse dans sa file d'attente LONDON_reply.

5. Sur LONDON , placez des messages dans la file d'attente de cluster.

- a) Définissez une file d'attente locale appelée LONDON_reply.
- b) Définissez les options MQOPEN sur MQOO_OUTPUT.
- c) Emettez l'appel MQOPEN pour ouvrir la file d'attente INVENTQ.
- d) Définissez le nom *ReplyToQ* dans le descripteur de message sur LONDON_reply.
- e) Emettez l'appel MQPUT pour insérer le message.
- f) Validez le message.

6. Sur NEWYORK , recevez le message dans la file d'attente de cluster et placez une réponse dans la file d'attente de réponses.

- a) Définissez les options MQOPEN sur MQOO_BROWSE.
- b) Emettez l'appel MQOPEN pour ouvrir la file d'attente INVENTQ.
- c) Emettez l'appel MQGET pour extraire le message de INVENTQ.
- d) Extrayez le nom *ReplyToQ* du descripteur de message.
- e) Placez le nom *ReplyToQ* dans la zone *ObjectName* du descripteur d'objet.
- f) Définissez les options MQOPEN sur MQOO_OUTPUT.
- g) Emettez l'appel MQOPEN pour ouvrir LONDON_reply sur le gestionnaire de files d'attente LONDON.
- h) Emettez l'appel MQPUT pour insérer le message dans LONDON_reply.

7. Sur LONDON , recevez la réponse.

- a) Définissez les options MQOPEN sur MQOO_BROWSE.
- b) Emettez l'appel MQOPEN pour ouvrir la file d'attente LONDON_reply.
- c) Emettez l'appel MQGET pour obtenir le message de LONDON_reply.

Ajout d'un gestionnaire de files d'attente à un cluster

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages vers les files d'attente de cluster et les rubriques sont transférés à l'aide de la file d'attente de transmission de cluster unique SYSTEM.CLUSTER.TRANSMIT.QUEUE.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY est configuré comme décrit dans [«Configuration d'un nouveau cluster»](#), à la page 331. Il contient deux gestionnaires de files d'attente, LONDON et NEWYORK, qui contiennent tous deux des référentiels complets.
- Le gestionnaire de files d'attente PARIS appartient à l'installation principale. Si ce n'est pas le cas, vous devez exécuter la commande **setmqenv** afin de configurer l'environnement de commande pour l'installation à laquelle appartient PARIS .
- La connectivité TCP existe entre les trois systèmes et le gestionnaire de files d'attente est configuré avec un programme d'écoute TCP qui démarre sous le contrôle du gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

1. Une nouvelle branche du magasin de chaînes est en cours de configuration à Paris et vous souhaitez ajouter un gestionnaire de files d'attente appelé PARIS au cluster.
2. Le gestionnaire de files d'attente PARIS envoie des mises à jour d'inventaire à l'application exécutée sur le système à New York en plaçant des messages dans la file d'attente INVENTQ .

Procédez comme suit pour ajouter un gestionnaire de files d'attente à un cluster.

Procédure

1. Choisissez le référentiel complet auquel PARIS fait référence en premier.

Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets. Il collecte des informations sur le cluster à partir d'un référentiel complet et génère donc son propre référentiel partiel. Choisissez l'un des référentiels comme référentiel complet. Dès qu'un nouveau gestionnaire de files d'attente est ajouté au cluster, il est immédiatement informé de l'autre référentiel. Les informations sur les modifications apportées à un gestionnaire de files d'attente sont envoyées directement à deux référentiels. Dans cet exemple, vous liez PARIS au gestionnaire de files d'attente LONDON, uniquement pour des raisons géographiques.


Remarque : Effectuez les étapes restantes dans n'importe quel ordre, après le démarrage du gestionnaire de files d'attente PARIS .

2. Définissez un canal CLUSRCVR sur le gestionnaire de files d'attente PARIS.

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal récepteur de cluster sur lequel il peut recevoir des messages. Sous PARIS, définissez:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

Le canal récepteur de cluster annonce la disponibilité du gestionnaire de files d'attente pour recevoir les messages des autres gestionnaires de files d'attente du cluster INVENTORY. Ne créez pas de définitions sur d'autres gestionnaires de files d'attente pour une extrémité d'envoi au canal récepteur de cluster INVENTORY . PARIS. D'autres définitions sont effectuées automatiquement lorsque cela est nécessaire. Voir [Canaux de cluster](#).

3.  Démarrez l'initiateur de canal sur IBM MQ for z/OS.
4. Définissez un canal CLUSSDR sur le gestionnaire de files d'attente PARIS.

Lorsque vous ajoutez à un cluster un gestionnaire de files d'attente qui n'est pas un référentiel complet, vous définissez un seul canal émetteur de cluster pour établir une connexion initiale à un référentiel complet. Voir [Canal émetteur de cluster: CLUSSDR](#) .

Sous PARIS, définissez la définition suivante pour un canal CLUSSDR appelé INVENTORY . LONDON sur le gestionnaire de files d'attente avec l'adresse réseau LONDON . CHSTORE . COM.

```
DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

5. Facultatif : Si vous ajoutez à un cluster un gestionnaire de files d'attente qui a été précédemment supprimé du même cluster, vérifiez qu'il s'affiche désormais en tant que membre de cluster. Si ce n'est pas le cas, effectuez les étapes supplémentaires suivantes:
 - a) Exécutez la commande **REFRESH CLUSTER** sur le gestionnaire de files d'attente que vous ajoutez. Cette étape arrête les canaux de cluster et fournit à votre cache de cluster local un nouvel ensemble de numéros de séquence qui sont assurés d'être à jour dans le reste du cluster.


```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

Remarque : Pour les clusters de grande taille, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster lors de son exécution, ainsi qu'ensuite, à des intervalles de 27 jours, lorsque les objets du cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir [L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster.](#)

- b) Redémarrez le canal CLUSSDR
(par exemple, à l'aide de la commande `START CHANNEL`).
- c) Redémarrez le canal CLUSRCVR.

Résultats

La figure suivante montre le cluster configuré par cette tâche.

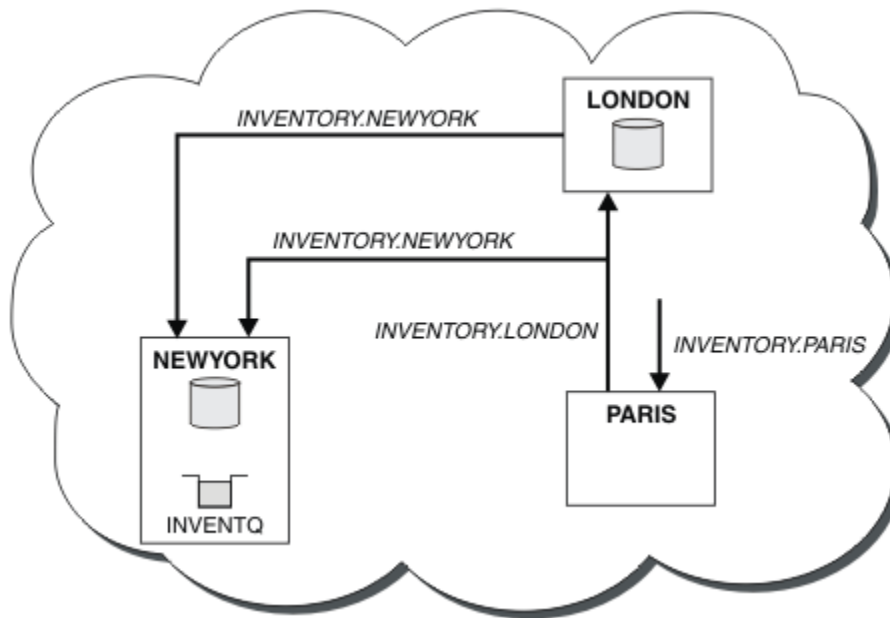


Figure 39. Le cluster INVENTORY avec trois gestionnaires de files d'attente

En n'effectuant que deux définitions, une définition CLUSRCVR et une définition CLUSSDR , nous avons ajouté le gestionnaire de files d'attente PARIS au cluster.

A présent, le gestionnaire de files d'attente PARIS apprend, à partir du référentiel complet à l'adresse LONDON, que la file d'attente INVENTQ est hébergée par le gestionnaire de files d'attente NEWYORK. Lorsqu'une application hébergée par le système à Paris tente d'insérer des messages dans le INVENTQ, PARIS définit automatiquement un canal émetteur de cluster pour se connecter au canal récepteur de cluster INVENTORY . NEWYORK. L'application peut recevoir des réponses lorsque son nom de gestionnaire de files d'attente est spécifié en tant que gestionnaire de files d'attente cible et qu'une file d'attente de réponse est fournie.

Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Avant de commencer

- Le gestionnaire de files d'attente n'est membre d'aucun cluster.
- Le cluster existe ; il existe un référentiel complet auquel ce gestionnaire de files d'attente peut se connecter directement et le référentiel est disponible. Pour la procédure de création du cluster, voir [«Configuration d'un nouveau cluster»](#), à la page 331.

Pourquoi et quand exécuter cette tâche

Cette tâche est une alternative à [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343, dans laquelle vous ajoutez un gestionnaire de files d'attente à un cluster qui place des messages de cluster dans une file d'attente de transmission unique.

Dans cette tâche, vous ajoutez un gestionnaire de files d'attente à un cluster qui crée automatiquement des files d'attente de transmission de cluster distinctes pour chaque canal émetteur de cluster.

Pour limiter le nombre de définitions de files d'attente, la valeur par défaut consiste à utiliser une seule file d'attente de transmission. L'utilisation de files d'attente de transmission distinctes est avantageuse si vous souhaitez surveiller le trafic destiné à différents gestionnaires de files d'attente et à différents clusters. Vous pouvez également séparer le trafic vers différentes destinations pour atteindre des objectifs d'isolement ou de performances.

Procédure

1. Modifiez le type de file d'attente de transmission du canal de cluster par défaut.

Modifiez le gestionnaire de files d'attente PARIS:

```
ALTER QMGR DEFCLXQ(CHANNEL)
```

Chaque fois que le gestionnaire de files d'attente crée un canal émetteur de cluster pour envoyer un message à un gestionnaire de files d'attente, il crée une file d'attente de transmission de cluster. La file d'attente de transmission est utilisée uniquement par ce canal émetteur de cluster. La file d'attente de transmission est permanente-dynamique. Il est créé à partir de la file d'attente modèle, SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE, avec le nom SYSTEM . CLUSTER . TRANSMIT . *ChannelName*.



Avertissement : Si vous utilisez SYSTEM . CLUSTER . TRANSMIT . QUEUES avec un gestionnaire de files d'attente mis à niveau à partir d'une version du produit antérieure à IBM WebSphere MQ 7.5, vérifiez que SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE a l'option [SHARE/NOSHARE](#) définie sur **SHARE**.

2. Choisissez le référentiel complet auquel PARIS fait référence en premier.

Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets. Il collecte des informations sur le cluster à partir d'un référentiel complet et génère donc son propre référentiel partiel. Choisissez l'un des référentiels comme référentiel complet. Dès qu'un nouveau gestionnaire de files d'attente est ajouté au cluster, il est immédiatement informé de l'autre référentiel. Les informations sur les modifications apportées à un gestionnaire de files d'attente sont envoyées directement à deux référentiels. Dans cet exemple, vous liez PARIS au gestionnaire de files d'attente LONDON, uniquement pour des raisons géographiques.

Remarque : Effectuez les étapes restantes dans n'importe quel ordre, après le démarrage du gestionnaire de files d'attente PARIS .

3. Définissez un canal CLUSRCVR sur le gestionnaire de files d'attente PARIS.

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal récepteur de cluster sur lequel il peut recevoir des messages. Sous PARIS, définissez:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
```

```

CONNNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')

```

Le canal récepteur de cluster annonce la disponibilité du gestionnaire de files d'attente pour recevoir les messages des autres gestionnaires de files d'attente du cluster INVENTORY. Ne créez pas de définitions sur d'autres gestionnaires de files d'attente pour une extrémité d'envoi au canal récepteur de cluster INVENTORY . PARIS. D'autres définitions sont effectuées automatiquement lorsque cela est nécessaire. Voir [Canaux de cluster](#).

4. Définissez un canal CLUSSDR sur le gestionnaire de files d'attente PARIS.

Lorsque vous ajoutez à un cluster un gestionnaire de files d'attente qui n'est pas un référentiel complet, vous définissez un seul canal émetteur de cluster pour établir une connexion initiale à un référentiel complet. Voir [Canal émetteur de cluster: CLUSSDR](#) .

Sous PARIS, définissez la définition suivante pour un canal CLUSSDR appelé INVENTORY . LONDON sur le gestionnaire de files d'attente avec l'adresse réseau LONDON . CHSTORE . COM.

```

DEFINE CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')

```

Le gestionnaire de files d'attente crée automatiquement la file d'attente de transmission de cluster dynamique permanente SYSTEM . CLUSTER . TRANSMIT . INVENTORY . LONDON à partir de la file d'attente modèle SYSTEM . CLUSTER . TRANSMIT . MODEL . QUEUE. Il définit l'attribut CLCHNAME de la file d'attente de transmission sur INVENTORY . LONDON.

Résultats

La figure suivante montre le cluster configuré par cette tâche.

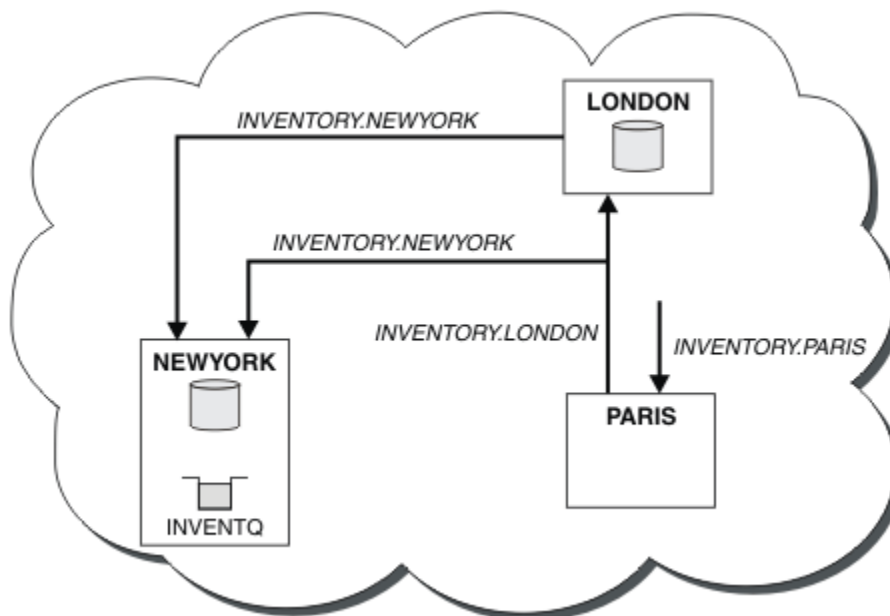


Figure 40. Le cluster INVENTORY avec trois gestionnaires de files d'attente

En n'effectuant que deux définitions, une définition CLUSRCVR et une définition CLUSSDR , nous avons ajouté le gestionnaire de files d'attente PARIS au cluster.

A présent, le gestionnaire de files d'attente PARIS apprend, à partir du référentiel complet à l'adresse LONDON, que la file d'attente INVENTQ est hébergée par le gestionnaire de files d'attente NEWYORK. Lorsqu'une application hébergée par le système à Paris tente d'insérer des messages dans le INVENTQ, PARIS définit automatiquement un canal émetteur de cluster pour se connecter au canal récepteur de cluster INVENTORY . NEWYORK. L'application peut recevoir des réponses lorsque son nom de gestionnaire

de files d'attente est spécifié en tant que gestionnaire de files d'attente cible et qu'une file d'attente de réponse est fournie.

Concepts associés

Comment choisir le type de file d'attente de transmission de cluster à utiliser

Tâches associées

Ajout d'un gestionnaire de files d'attente à un cluster à l'aide de DHCP

Ajoutez un gestionnaire de files d'attente à un cluster à l'aide de DHCP. La tâche illustre l'omission de la valeur CONNAME dans une définition CLUSRCVR .

Ajout d'un gestionnaire de files d'attente à un cluster à l'aide de DHCP


Ajoutez un gestionnaire de files d'attente à un cluster à l'aide de DHCP. La tâche illustre l'omission de la valeur CONNAME dans une définition CLUSRCVR .

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

La tâche illustre deux fonctions spéciales:

- Possibilité d'omettre la valeur CONNAME dans une définition CLUSRCVR .
- Possibilité d'utiliser +QMNAME+ sur une définition CLUSSDR .

 Aucune des deux fonctions n'est fournie dans z/OS.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans «[Configuration d'un nouveau cluster](#)», à la [page 331](#). Il contient deux gestionnaires de files d'attente, LONDON et NEWYORK, qui contiennent tous deux des référentiels complets.
- Une nouvelle branche du magasin de chaînes est en cours de configuration à Paris et vous souhaitez ajouter un gestionnaire de files d'attente appelé PARIS au cluster.
- Le gestionnaire de files d'attente PARIS envoie des mises à jour d'inventaire à l'application exécutée sur le système à New York en plaçant des messages dans la file d'attente INVENTQ.
- Il existe une connectivité réseau entre les trois systèmes.
- Le protocole réseau est TCP.
- Le système de gestionnaire de files d'attente PARIS utilise DHCP, ce qui signifie que les adresses IP peuvent changer au redémarrage du système.
- Les canaux entre les systèmes PARIS et LONDON sont nommés selon une convention de dénomination définie. La convention utilise le nom de gestionnaire de files d'attente du gestionnaire de files d'attente de référentiel complet sous LONDON.
- Les administrateurs du gestionnaire de files d'attente PARIS ne disposent pas d'informations sur le nom du gestionnaire de files d'attente dans le référentiel LONDON . Le nom du gestionnaire de files d'attente dans le référentiel LONDON est susceptible d'être modifié.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour ajouter un gestionnaire de files d'attente à un cluster à l'aide de DHCP.

Procédure

1. Choisissez le référentiel complet auquel PARIS fait référence en premier.

Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets. Il collecte des informations sur le cluster à partir d'un référentiel complet et génère donc son propre référentiel partiel. Choisissez l'un des référentiels comme référentiel complet. Dès qu'un

nouveau gestionnaire de files d'attente est ajouté au cluster, il est immédiatement informé de l'autre référentiel. Les informations sur les modifications apportées à un gestionnaire de files d'attente sont envoyées directement à deux référentiels. Dans cet exemple, nous choisissons de lier PARIS au gestionnaire de files d'attente LONDON, uniquement pour des raisons géographiques.

Remarque : Effectuez les étapes restantes dans n'importe quel ordre, après le démarrage du gestionnaire de files d'attente PARIS .

2. Définissez un canal CLUSRCVR sur le gestionnaire de files d'attente PARIS.

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal récepteur de cluster sur lequel il peut recevoir des messages. Sous PARIS, définissez:

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSRCVR)
TRPTYPE(TCP) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager PARIS')
```

Le canal récepteur de cluster annonce la disponibilité du gestionnaire de files d'attente pour recevoir les messages des autres gestionnaires de files d'attente du cluster INVENTORY. Vous n'avez pas besoin de spécifier CONNAME sur le canal récepteur de cluster. Vous pouvez demander à IBM MQ de trouver le nom de connexion à partir du système, soit en omettant CONNAME, soit en spécifiant `CONNAME('')`. IBM MQ génère la valeur CONNAME à l'aide de l'adresse IP actuelle du système ; voir [CONNAME](#) . Il n'est pas nécessaire de créer des définitions sur d'autres gestionnaires de files d'attente pour une extrémité d'envoi au canal récepteur de cluster INVENTORY . PARIS. D'autres définitions sont effectuées automatiquement lorsque cela est nécessaire.

3. Définissez un canal CLUSSDR sur le gestionnaire de files d'attente PARIS.

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal émetteur de cluster sur lequel il peut envoyer des messages à son référentiel complet initial. Sous PARIS, définissez la définition suivante pour un canal appelé INVENTORY . +QMNAME+ sur le gestionnaire de files d'attente avec l'adresse réseau LONDON . CHSTORE . COM.

```
DEFINE CHANNEL(INVENTORY.+QMNAME+) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(LONDON.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at LONDON')
```

4. Facultatif : Si vous ajoutez à un cluster un gestionnaire de files d'attente qui a été précédemment supprimé du même cluster, vérifiez qu'il s'affiche désormais en tant que membre de cluster. Si ce n'est pas le cas, effectuez les étapes supplémentaires suivantes:

- a) Exécutez la commande **REFRESH CLUSTER** sur le gestionnaire de files d'attente que vous ajoutez. Cette étape arrête les canaux de cluster et fournit à votre cache de cluster local un nouvel ensemble de numéros de séquence qui sont assurés d'être à jour dans le reste du cluster.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

Remarque : Pour les clusters de grande taille, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster lors de son exécution, ainsi qu'ensuite, à des intervalles de 27 jours, lorsque les objets du cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir [L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster](#).

- b) Redémarrez le canal CLUSSDR
(par exemple, à l'aide de la commande [START CHANNEL](#)).
- c) Redémarrez le canal CLUSRCVR.

Résultats

Le cluster configuré par cette tâche est le même que pour [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343:

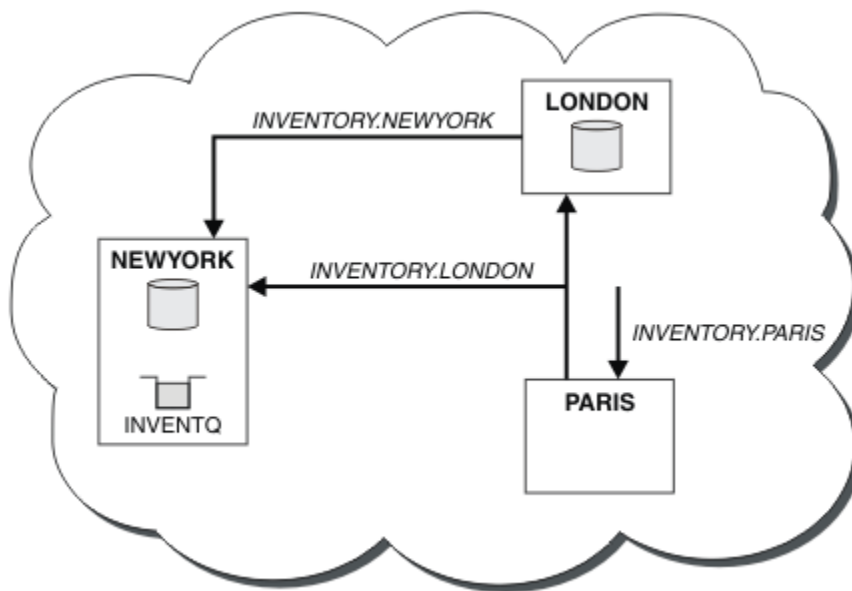


Figure 41. Cluster INVENTORY avec trois gestionnaires de files d'attente

En n'effectuant que deux définitions, une définition CLUSRCVR et une définition CLUSSDR , nous avons ajouté le gestionnaire de files d'attente PARIS au cluster.

Sur le gestionnaire de files d'attente PARIS , la commande CLUSSDR contenant la chaîne +QMNAME+ démarre. Sur le système LONDON , IBM MQ résout le +QMNAME+ en nom de gestionnaire de files d'attente (LONDON). IBM MQ fait ensuite correspondre la définition d'un canal appelé INVENTORY . LONDON à la définition CLUSRCVR correspondante.

IBM MQ renvoie le nom de canal résolu au gestionnaire de files d'attente PARIS . Dans PARIS, la définition de canal CLUSSDR pour le canal appelé INVENTORY . +QMNAME+ est remplacée par une définition CLUSSDR générée en interne pour INVENTORY . LONDON. Cette définition contient le nom de canal résolu, mais dans le cas contraire, elle est identique à la définition +QMNAME+ que vous avez créée. Les référentiels de cluster sont également mis à jour avec la définition de canal avec le nom de canal nouvellement résolu.

Remarque :

1. Le canal créé avec le nom +QMNAME+ devient inactif immédiatement. Il n'est jamais utilisé pour transmettre des données.
2. Les exits de canal peuvent voir le changement de nom de canal entre un appel et le suivant.

A présent, le gestionnaire de files d'attente PARIS apprend, à partir du référentiel situé dans LONDON, que la file d'attente INVENTQ est hébergée par le gestionnaire de files d'attente NEWYORK. Lorsqu'une application hébergée par le système à Paris tente d'insérer des messages dans le INVENTQ, PARIS , elle définit automatiquement un canal émetteur de cluster pour se connecter au canal récepteur de cluster INVENTORY . NEWYORK. L'application peut recevoir des réponses lorsque son nom de gestionnaire de files d'attente est spécifié en tant que gestionnaire de files d'attente cible et qu'une file d'attente de réponse est fournie.

Tâches associées

Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Référence associée

De la définition d'un canal

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente

Ajoutez un autre gestionnaire de files d'attente au cluster pour héberger une autre file d'attente INVENTQ . Les demandes sont envoyées alternativement aux files d'attente de chaque gestionnaire de files d'attente. Il n'est pas nécessaire d'apporter des modifications à l'hôte INVENTQ existant.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343. Il contient trois gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets, PARIS contient un référentiel partiel. L'application d'inventaire s'exécute sur le système à New York, connecté au gestionnaire de files d'attente NEWYORK . L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ .
- Un nouveau magasin est en train d'être créé à Toronto. Pour fournir une capacité supplémentaire, vous souhaitez exécuter l'application d'inventaire sur le système à Toronto et à New York.
- La connectivité du réseau existe entre les quatre systèmes.
- Le protocole réseau est TCP.

Remarque : Le gestionnaire de files d'attente TORONTO ne contient qu'un référentiel partiel. Si vous souhaitez ajouter un gestionnaire de files d'attente de référentiel complet à un cluster, voir [«Déplacement d'un référentiel complet vers un autre gestionnaire de files d'attente»](#), à la page 355.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour ajouter un gestionnaire de files d'attente qui héberge une file d'attente.

Procédure

1. Choisissez le référentiel complet auquel TORONTO fait référence en premier.

Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets. Il collecte des informations sur le cluster à partir d'un référentiel complet et génère donc son propre référentiel partiel. Il n'a pas de signification particulière pour le référentiel que vous choisissez. Dans cet exemple, nous choisissons NEWYORK. Une fois que le nouveau gestionnaire de files d'attente a rejoint le cluster, il communique avec les deux référentiels.

2. Définissez le canal CLUSRCVR .

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal récepteur de cluster sur lequel il peut recevoir des messages. Sous TORONTO, définissez un canal CLUSRCVR :

```
DEFINE CHANNEL(INVENTORY.TORONTO) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(TORONTO.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for TORONTO')
```

Le gestionnaire de files d'attente TORONTO annonce sa disponibilité pour recevoir des messages d'autres gestionnaires de files d'attente du cluster INVENTORY à l'aide de son canal récepteur de cluster.

3. Définissez un canal CLUSSDR sur le gestionnaire de files d'attente TORONTO.

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal émetteur de cluster sur lequel il peut envoyer des messages à son premier référentiel complet. Dans ce cas, choisissez NEWYORK. TORONTO a besoin de la définition suivante:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
```

```
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from TORONTO to repository at NEWYORK')
```

4. Facultatif : Si vous ajoutez à un cluster un gestionnaire de files d'attente qui a été précédemment supprimé du même cluster, vérifiez qu'il s'affiche désormais en tant que membre de cluster. Si ce n'est pas le cas, effectuez les étapes supplémentaires suivantes:

- a) Exécutez la commande **REFRESH CLUSTER** sur le gestionnaire de files d'attente que vous ajoutez. Cette étape arrête les canaux de cluster et fournit à votre cache de cluster local un nouvel ensemble de numéros de séquence qui sont assurés d'être à jour dans le reste du cluster.

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

Remarque : Pour les clusters de grande taille, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster lors de son exécution, ainsi qu'ensuite, à des intervalles de 27 jours, lorsque les objets du cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir [L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster](#).

- b) Redémarrez le canal CLUSSDR
(par exemple, à l'aide de la commande [START CHANNEL](#)).
 - c) Redémarrez le canal CLUSRCVR.
5. Recherchez les affinités de message dans l'application d'inventaire.

Avant de poursuivre, assurez-vous que l'application d'inventaire n'a pas de dépendances sur la séquence de traitement des messages et installez l'application sur le système à Toronto.

6. Définissez la file d'attente de cluster INVENTQ.

La file d'attente INVENTQ , qui est déjà hébergée par le gestionnaire de files d'attente NEWYORK , doit également être hébergée par TORONTO. Définissez-le sur le gestionnaire de files d'attente TORONTO comme suit:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

Résultats

La [Figure 42](#), à la page 353 présente le cluster INVENTORY configuré par cette tâche.

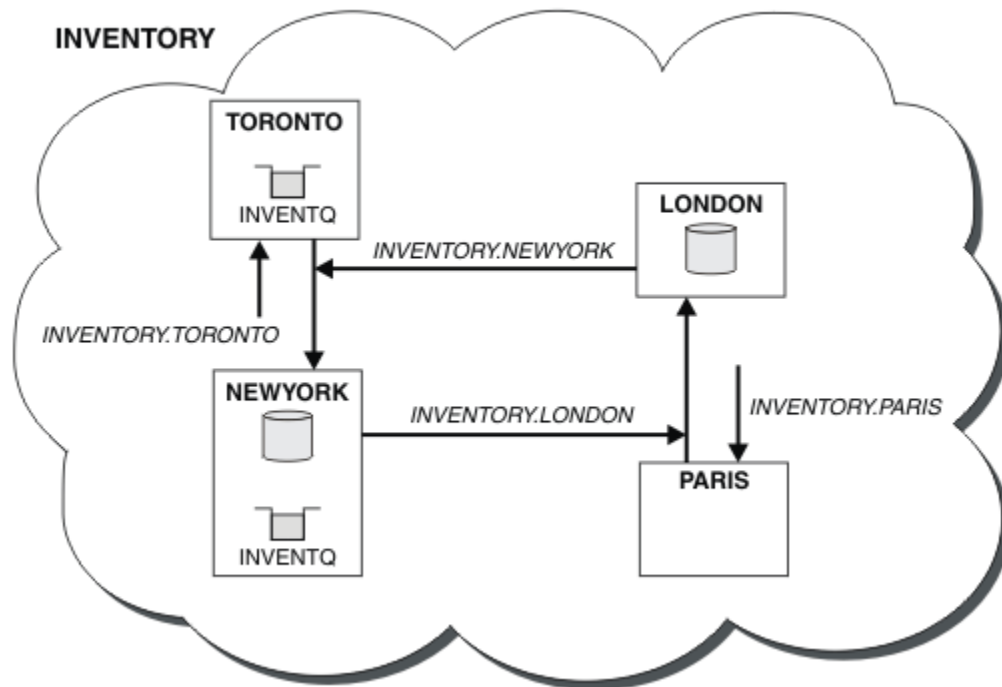


Figure 42. Cluster INVENTORY avec quatre gestionnaires de files d'attente

La file d'attente INVENTQ et l'application d'inventaire sont désormais hébergées sur deux gestionnaires de files d'attente du cluster. Cela augmente leur disponibilité, accélère le débit des messages et permet de répartir la charge de travail entre les deux gestionnaires de files d'attente. Les messages insérés dans INVENTQ par TORONTO ou NEWYORK sont traités par l'instance sur le gestionnaire de files d'attente local dans la mesure du possible. Les messages insérés par LONDON ou PARIS sont routés alternativement vers TORONTO ou NEWYORK, afin que la charge de travail soit équilibrée.

Cette modification du cluster a été effectuée sans que vous ayez à modifier les définitions sur les gestionnaires de files d'attente NEWYORK, LONDON et PARIS. Les référentiels complets de ces gestionnaires de files d'attente sont mis à jour automatiquement avec les informations dont ils ont besoin pour pouvoir envoyer des messages à INVENTQ à l'adresse TORONTO. L'application d'inventaire continue de fonctionner si l'un des gestionnaires de files d'attente NEWYORK ou TORONTO devient indisponible et qu'elle dispose d'une capacité suffisante. L'application d'inventaire doit pouvoir fonctionner correctement si elle est hébergée dans les deux emplacements.

Comme vous pouvez le voir dans le résultat de cette tâche, la même application peut s'exécuter sur plusieurs gestionnaires de files d'attente. Vous pouvez effectuer une mise en cluster pour répartir la charge de travail de manière uniforme.

Il se peut qu'une application ne puisse pas traiter les enregistrements dans les deux emplacements. Par exemple, supposons que vous décidiez d'ajouter une requête de compte client et une application de mise à jour s'exécutant dans LONDON et NEWYORK. Un enregistrement de compte ne peut être conservé qu'à un seul endroit. Vous pouvez décider de contrôler la distribution des demandes à l'aide d'une technique de partitionnement de données. Vous pouvez diviser la distribution des enregistrements. Vous pouvez faire en sorte que la moitié des enregistrements, par exemple les numéros de compte 00000 à 49999, soient conservés dans LONDON. L'autre moitié, comprise entre 50000 et 99999, est conservée dans NEWYORK. Vous pouvez ensuite écrire un programme d'exit de charge de travail de cluster pour examiner la zone de compte dans tous les messages et acheminer les messages vers le gestionnaire de files d'attente approprié.

Que faire ensuite

z/OS Maintenant que vous avez terminé toutes les définitions, si ce n'est pas déjà fait, démarrez l'initiateur de canal sur IBM MQ for z/OS.

Sur toutes les plateformes, démarrez un programme d'écoute sur le gestionnaire de files d'attente TORONTO. Le programme d'écoute attend les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

z/OS Adding a queue sharing group to existing clusters

Add a queue sharing group on z/OS to existing clusters.

Before you begin

Note:

1. For changes to a cluster to be propagated throughout the cluster, at least one full repository must always be available. Ensure that your repositories are available before starting this task.
2. Queue sharing groups are supported only on IBM MQ for z/OS. This task is not applicable to other platforms.

Scenario:

- The INVENTORY cluster has been set up as described in [“Configuration d'un nouveau cluster” on page 331](#). It contains two queue managers, LONDON and NEWYORK.
- You want to add a queue sharing group to this cluster. The group, QSGP, comprises three queue managers, P1, P2, and P3. They share an instance of the INVENTQ queue, which is to be defined by P1.

About this task

Follow these steps to add new queue managers that host a shared queue.

Procedure

1. Decide which full repository the queue managers refer to first.

Every queue manager in a cluster must refer to one or other of the full repositories. It gathers information about the cluster from a full repository and so builds up its own partial repository. It is of no particular significance which full repository you choose. In this example, choose NEWYORK. Once the queue sharing group has joined the cluster, it communicates with both of the full repositories.

2. Define the CLUSRCVR channels.

Every queue manager in a cluster needs to define a cluster-receiver channel on which it can receive messages. On P1, P2, and P3, define:

```
DEFINE CHANNEL(INVENTORY.Pn) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(Pn.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for sharing queue manager')
```

The cluster-receiver channel advertises the availability of each queue manager to receive messages from other queue managers in the cluster INVENTORY.

3. Define a CLUSSDR channel for the queue sharing group.

Every member of a cluster needs to define one cluster-sender channel on which it can send messages to its first full repository. In this case we have chosen NEWYORK. One of the queue managers in the queue sharing group needs the following group definition. The definition ensures that every queue manager has a cluster-sender channel definition.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
```

```
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) QSGDISP(GROUP)
DESCR('Cluster-sender channel to repository at NEWYORK')
```

4. Define the shared queue.

Define the queue INVENTQ on P1 as follows:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY) QSGDISP(SHARED) CFSTRUCT(STRUCTURE)
```

Start the channel initiator and a listener program on the new queue manager. The listener program listens for incoming network requests and starts the cluster-receiver channel when it is needed.

Results

Figure 43 on page 355 shows the cluster set up by this task.

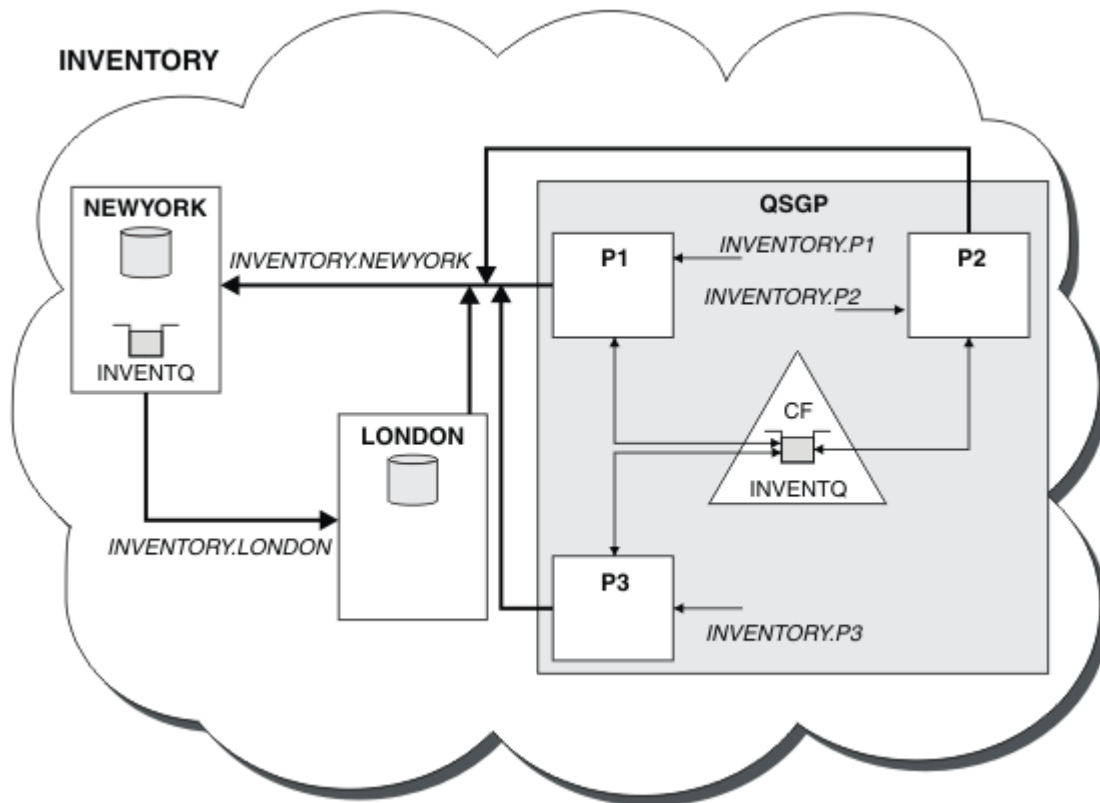


Figure 43. Cluster and queue sharing group

Now messages put on the INVENTQ queue by LONDON are routed alternately around the four queue managers advertised as hosting the queue.

What to do next

A benefit of having members of a queue sharing group host a cluster queue is any member of the group can reply to a request. In this case perhaps P1 becomes unavailable after receiving a message on the shared queue. Another member of the queue sharing group can reply instead.

Déplacement d'un référentiel complet vers un autre gestionnaire de files d'attente

Déplacez un référentiel complet d'un gestionnaire de files d'attente vers un autre, en créant le nouveau référentiel à partir des informations contenues dans le deuxième référentiel.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343.
- Pour des raisons métier, vous souhaitez maintenant supprimer le référentiel complet du gestionnaire de files d'attente LONDON et le remplacer par un référentiel complet dans le gestionnaire de files d'attente PARIS. Le gestionnaire de files d'attente NEWYORK doit continuer à conserver un référentiel complet.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour déplacer un référentiel complet vers un autre gestionnaire de files d'attente.

Procédure

1. Modifiez PARIS pour en faire un gestionnaire de files d'attente de référentiel complet.

Sous PARIS, exécutez la commande suivante:

```
ALTER QMGR REPOS(INVENTORY)
```

2. Ajout d'un canal CLUSSDR sur PARIS

PARIS dispose actuellement d'un canal émetteur de cluster pointant vers LONDON. LONDON ne doit plus contenir de référentiel complet pour le cluster. PARIS doit disposer d'un nouveau canal émetteur de cluster qui pointe vers NEWYORK, où se trouve désormais l'autre référentiel complet.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from PARIS to repository at NEWYORK')
```

3. Définissez un canal CLUSSDR sur NEWYORK qui pointe vers PARIS

Actuellement, NEWYORK possède un canal émetteur de cluster pointant vers LONDON. Maintenant que l'autre référentiel complet a été déplacé vers PARIS, vous devez ajouter un nouveau canal émetteur de cluster à NEWYORK qui pointe vers PARIS.

```
DEFINE CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNNAME(PARIS.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from NEWYORK to repository at PARIS')
```

Lorsque vous ajoutez le canal émetteur de cluster à PARIS, PARIS apprend sur le cluster à partir de NEWYORK. Il crée son propre référentiel complet à l'aide des informations de NEWYORK.

4. Vérifiez que le gestionnaire de files d'attente PARIS dispose désormais d'un référentiel complet

Vérifiez que le gestionnaire de files d'attente PARIS a généré son propre référentiel complet à partir du référentiel complet sur le gestionnaire de files d'attente NEWYORK. Exécutez les commandes suivantes :

```
DIS QCLUSTER(*) CLUSTER (INVENTORY)
DIS CLUSQMGR(*) CLUSTER (INVENTORY)
```

Vérifiez que ces commandes affichent les détails des mêmes ressources dans ce cluster que sur NEWYORK.

Remarque : Si le gestionnaire de files d'attente NEWYORK n'est pas disponible, cette génération d'informations ne peut pas aboutir. Ne passez pas à l'étape suivante tant que la tâche n'est pas terminée.

5. Modification de la définition de gestionnaire de files d'attente sous LONDON

Enfin, modifiez le gestionnaire de files d'attente dans LONDON de sorte qu'il ne stocke plus de référentiel complet pour le cluster. Sous LONDON, exécutez la commande suivante:

```
ALTER QMGR REPOS(' ')
```

Le gestionnaire de files d'attente ne reçoit plus d'informations de cluster. Au bout de 30 jours, les informations stockées dans son référentiel complet expirent. Le gestionnaire de files d'attente LONDON génère désormais son propre référentiel partiel.

6. Supprimez ou modifiez les définitions en attente.

Lorsque vous êtes certain que la nouvelle disposition de votre cluster fonctionne comme prévu, supprimez ou modifiez les définitions CLUSSDR définies manuellement qui ne sont plus correctes.

- Sur le gestionnaire de files d'attente PARIS , vous devez arrêter et supprimer le canal émetteur de cluster vers LONDON, puis émettre la commande de démarrage de canal afin que le cluster puisse à nouveau utiliser les canaux automatiques:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
START CHANNEL(INVENTORY.LONDON)
```

- Sur le gestionnaire de files d'attente NEWYORK , vous devez arrêter et supprimer le canal émetteur de cluster vers LONDON, puis émettre la commande de démarrage de canal afin que le cluster puisse à nouveau utiliser les canaux automatiques:

```
STOP CHANNEL(INVENTORY.LONDON)
DELETE CHANNEL(INVENTORY.LONDON)
START CHANNEL(INVENTORY.LONDON)
```

- Remplacez tous les autres canaux émetteurs de cluster définis manuellement qui pointent vers LONDON sur tous les gestionnaires de files d'attente du cluster par des canaux qui pointent vers NEWYORK ou PARIS. Après avoir supprimé un canal, émettez toujours la commande **start channel** pour que le cluster puisse à nouveau utiliser les canaux automatiques. Dans ce petit exemple, il n'y en a pas d'autres. Pour vérifier s'il existe d'autres éléments que vous avez oubliés, exécutez la commande DISPLAY CHANNEL à partir de chaque gestionnaire de files d'attente en spécifiant TYPE(CLUSSDR). Exemple :

```
DISPLAY CHANNEL(*) TYPE(CLUSSDR)
```

Il est important que vous effectuiez cette tâche dès que possible après avoir déplacé le référentiel complet de LONDON vers PARIS. Avant d'effectuer cette tâche, les gestionnaires de files d'attente qui ont défini manuellement des canaux CLUSSDR nommés INVENTORY.LONDON peuvent envoyer des demandes d'informations à l'aide de ce canal.

Une fois que LONDON a cessé d'être un référentiel complet, s'il reçoit de telles demandes, il écrit des messages d'erreur dans son journal des erreurs du gestionnaire de files d'attente. Les exemples suivants montrent quels messages d'erreur peuvent être affichés sous LONDON:

- AMQ9428: Unexpected publication of a cluster queue object received
- AMQ9432: Query received by a non-repository queue manager

Le gestionnaire de files d'attente LONDON ne répond pas aux demandes d'informations car il ne s'agit plus d'un référentiel complet. Les gestionnaires de files d'attente qui demandent des informations à LONDON doivent s'appuyer sur NEWYORK pour les informations de cluster jusqu'à ce que leurs

définitions CLUSSDR définies manuellement soient corrigées pour pointer vers PARIS. Cette situation ne doit pas être tolérée comme une configuration valide à long terme.

Résultats

La [Figure 44](#), à la page 358 présente le cluster configuré par cette tâche.

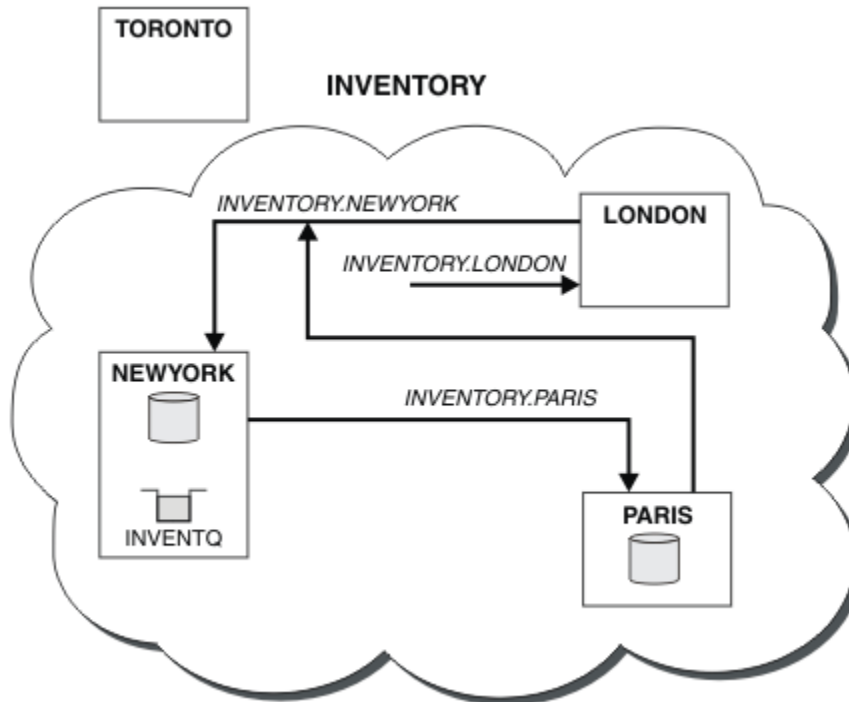


Figure 44. Le cluster INVENTORY avec le référentiel complet déplacé vers PARIS

Conversion d'un réseau existant en cluster

Convertissez un réseau de mise en file d'attente répartie existant en un cluster et ajoutez un gestionnaire de files d'attente supplémentaire pour augmenter la capacité.

Avant de commencer

Dans «[Configuration d'un nouveau cluster](#)», à la page 331 via «[Déplacement d'un référentiel complet vers un autre gestionnaire de files d'attente](#)», à la page 355, vous avez créé et étendu un nouveau cluster. Les deux tâches suivantes explorent une approche différente: celle de la conversion d'un réseau existant de gestionnaires de files d'attente en cluster.

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Un réseau IBM MQ est déjà en place, connectant les branches nationales d'un magasin en chaîne. Il possède une structure en étoile: tous les gestionnaires de files d'attente sont connectés à un gestionnaire de files d'attente central. Le gestionnaire de files d'attente central se trouve sur le système sur lequel l'application d'inventaire s'exécute. L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ, pour laquelle chaque gestionnaire de files d'attente possède une définition de file d'attente éloignée.

Ce réseau est illustré dans la [Figure 45](#), à la page 359.

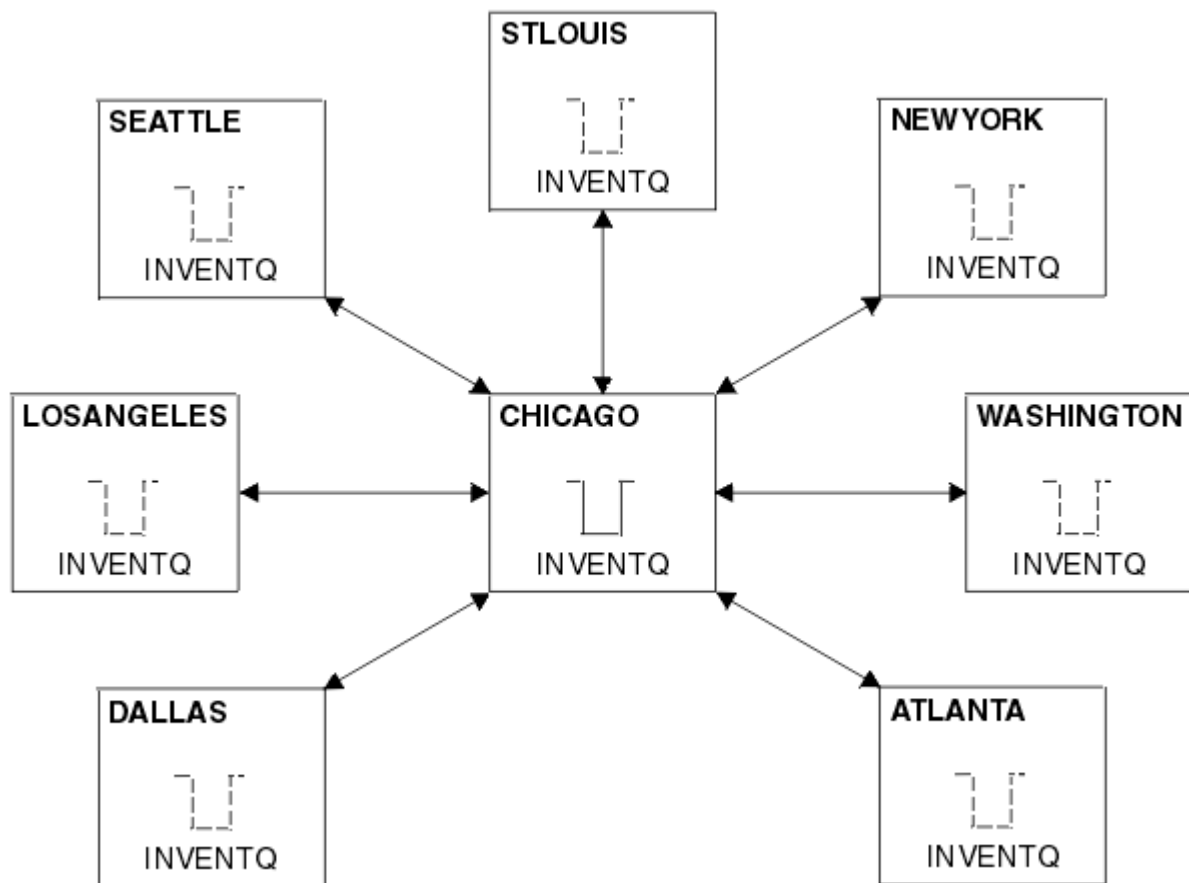


Figure 45. Un réseau concentrateur et satellite

- Pour faciliter l'administration, vous allez convertir ce réseau en cluster et créer un autre gestionnaire de files d'attente sur le site central pour partager la charge de travail.

Le nom du cluster est CHNSTORE.

Remarque : Le nom de cluster CHNSTORE a été sélectionné pour permettre la création de noms de canal récepteur de cluster à l'aide de noms au format `cluster_name.queue_manager_name` qui ne dépassent pas la longueur maximale de 20 caractères, par exemple CHNSTORE.WASHINGTON.

- Les deux gestionnaires de files d'attente centraux doivent héberger des référentiels complets et être accessibles à l'application d'inventaire.
- L'application d'inventaire doit être gérée par l'arrivée de messages dans la file d'attente INVENTQ hébergée par l'un des gestionnaires de files d'attente centraux.
- L'application d'inventaire doit être la seule application exécutée en parallèle et accessible par plusieurs gestionnaires de files d'attente. Toutes les autres applications continuent de s'exécuter comme auparavant.
- Toutes les branches ont une connectivité réseau aux deux gestionnaires de files d'attente centraux.
- Le protocole réseau est TCP.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour convertir un réseau existant en cluster.

Procédure

1. Recherchez les affinités de message dans l'application d'inventaire.

Avant de continuer, assurez-vous que l'application peut gérer les affinités de message. Les affinités de message sont la relation entre des messages conversationnels échangés entre deux applications,

où les messages doivent être traités par un gestionnaire de files d'attente particulier ou dans une séquence particulière. Pour plus d'informations sur les affinités de message, voir «[Traitement des affinités de message](#)», à la page 442

2. Modifiez les deux gestionnaires de files d'attente centraux pour qu'ils soient des gestionnaires de files d'attente de référentiel complet.

Les deux gestionnaires de files d'attente CHICAGO et CHICAGO2 se trouvent sur le concentrateur de ce réseau. Vous avez décidé de concentrer toutes les activités associées au cluster de magasins de chaîne sur ces deux gestionnaires de files d'attente. Outre l'application d'inventaire et les définitions de la file d'attente INVENTQ, vous souhaitez que ces gestionnaires de files d'attente hébergent les deux référentiels complets du cluster. Sur chacun des deux gestionnaires de files d'attente, exécutez la commande suivante:

```
ALTER QMGR REPOS(CHNSTORE)
```

3. Définissez un canal CLUSRCVR sur chaque gestionnaire de files d'attente.

Sur chaque gestionnaire de files d'attente du cluster, définissez un canal récepteur de cluster et un canal émetteur de cluster. Peu importe le canal que vous définissez en premier.

Créez une définition CLUSRCVR pour annoncer au cluster chaque gestionnaire de files d'attente, son adresse réseau et d'autres informations. Par exemple, sur le gestionnaire de files d'attente ATLANTA:

```
DEFINE CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(ATLANTA.CHSTORE.COM) CLUSTER(CHNSTORE)
DESCR('Cluster-receiver channel')
```

4. Définissez un canal CLUSSDR sur chaque gestionnaire de files d'attente

Créez une définition CLUSSDR sur chaque gestionnaire de files d'attente pour lier ce gestionnaire de files d'attente à l'un ou l'autre des gestionnaires de files d'attente de référentiel complet. Par exemple, vous pouvez lier ATLANTA à CHICAGO2:

```
DEFINE CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO2.CHSTORE.COM) CLUSTER(CHNSTORE)
DESCR('Cluster-sender channel to repository queue manager')
```

5. Installez l'application d'inventaire sur CHICAGO2.

Vous disposez déjà de l'application d'inventaire sur le gestionnaire de files d'attente CHICAGO. Vous devez maintenant effectuer une copie de cette application sur le gestionnaire de files d'attente CHICAGO2.


6. Définissez la file d'attente INVENTQ sur les gestionnaires de files d'attente centraux.

Sous CHICAGO, modifiez la définition de file d'attente locale pour la file d'attente INVENTQ afin que la file d'attente soit disponible pour le cluster. Entrez la commande suivante :

```
ALTER QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

Sous CHICAGO2, définissez la même file d'attente:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(CHNSTORE)
```

 Sous z/OS, vous pouvez utiliser l'option MAKEDEF de la fonction COMMAND de **CSQUITL** pour effectuer une copie exacte sous CHICAGO2 de INVENTQ sous CHICAGO.

Lorsque vous créez ces définitions, un message est envoyé aux référentiels complets à l'adresse CHICAGO et CHICAGO2 et les informations qu'ils contiennent sont mises à jour. Le gestionnaire de files d'attente détecte dans les référentiels complets lorsqu'il insère un message dans INVENTQ, qu'il existe un choix de destinations pour les messages.

7. Vérifiez que les modifications apportées au cluster ont été propagées.

Vérifiez que les définitions que vous avez créées à l'étape précédente ont été propagées via le cluster. Exécutez la commande suivante sur un gestionnaire de files d'attente de référentiel complet:

```
DIS QCLUSTER(INVENTQ)
```

Ajout d'un nouveau cluster interconnecté

Ajoutez un nouveau cluster qui partage certains gestionnaires de files d'attente avec un cluster existant.

Avant de commencer

Remarque :

1. Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.
2. Avant de commencer cette tâche, recherchez les conflits de noms de file d'attente et comprenez les conséquences. Vous devrez peut-être renommer une file d'attente ou définir des alias de file d'attente avant de continuer.

Scénario :

- Un cluster IBM MQ a été configuré comme décrit dans [«Conversion d'un réseau existant en cluster»](#), à la page 358.
- Un nouveau cluster appelé MAILORDER doit être implémenté. Ce cluster comprend quatre des gestionnaires de files d'attente qui se trouvent dans le cluster CHNSTORE ; CHICAGO, CHICAG02, SEATTLE et ATLANTA, et deux gestionnaires de files d'attente supplémentaires ; HARTFORD et OMAHA. L'application MAILORDER s'exécute sur le système à Omaha, connectée au gestionnaire de files d'attente OMAHA. Il est géré par les autres gestionnaires de files d'attente du cluster qui placent des messages dans la file d'attente MORDERQ .
- Les référentiels complets du cluster MAILORDER sont gérés sur les deux gestionnaires de files d'attente CHICAGO et CHICAG02.
- Le protocole réseau est TCP.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour ajouter un nouveau cluster interconnecté.

Procédure

1. Créez une liste de noms des noms de cluster.

Les gestionnaires de files d'attente de référentiel complet dans CHICAGO et CHICAG02 vont maintenant contenir les référentiels complets pour les deux clusters CHNSTORE et MAILORDER. Créez d'abord une liste de noms contenant les noms des clusters. Définissez la liste de noms sous CHICAGO et CHICAG02, comme suit:

```
DEFINE NAMELIST(CHAINMAIL)
DESCR('List of cluster names')
NAMES(CHNSTORE, MAILORDER)
```

2. Modifiez les deux définitions de gestionnaire de files d'attente.

A présent, modifiez les deux définitions de gestionnaire de files d'attente dans CHICAGO et CHICAG02. Actuellement, ces définitions indiquent que les gestionnaires de files d'attente contiennent des référentiels complets pour le cluster CHNSTORE. Modifiez cette définition pour indiquer que les gestionnaires de files d'attente contiennent des référentiels complets pour tous

les clusters répertoriés dans la liste de noms CHAINMAIL . Modifiez les définitions de gestionnaire de files d'attente CHICAGO et CHICAGO2 :

```
ALTER QMGR REPOS(' ') REPOSNL(CHAINMAIL)
```

3. Modifiez les canaux CLUSRCVR sur CHICAGO et CHICAGO2.

Les définitions de canal CLUSRCVR dans CHICAGO et CHICAGO2 indiquent que les canaux sont disponibles dans le cluster CHNSTORE. Vous devez modifier la définition du récepteur de cluster pour indiquer que les canaux sont disponibles pour tous les clusters répertoriés dans la liste de noms CHAINMAIL . Modifiez la définition du récepteur de cluster dans CHICAGO:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR)  
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Dans CHICAGO2, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSRCVR)  
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

4. Modifiez les canaux CLUSSDR sur CHICAGO et CHICAGO2.

Modifiez les deux définitions de canal CLUSSDR pour ajouter la liste de noms. Dans CHICAGO, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)  
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Dans CHICAGO2, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)  
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

5. Créez une liste de noms sur SEATTLE et ATLANTA.

Etant donné que SEATTLE et ATLANTA vont être membres de plusieurs clusters, vous devez créer une liste de noms contenant les noms des clusters. Définissez la liste de noms sous SEATTLE et ATLANTA, comme suit:

```
DEFINE NAMELIST(CHAINMAIL)  
DESCR('List of cluster names')  
NAMES(CHNSTORE, MAILORDER)
```

6. Modifiez les canaux CLUSRCVR sur SEATTLE et ATLANTA.

Les définitions de canal CLUSRCVR dans SEATTLE et ATLANTA indiquent que les canaux sont disponibles dans le cluster CHNSTORE. Modifiez les définitions de canal de réception de cluster pour indiquer que les canaux sont disponibles pour tous les clusters répertoriés dans la liste de noms CHAINMAIL . Dans SEATTLE, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.SEATTLE) CHLTYPE(CLUSRCVR)  
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Dans ATLANTA, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.ATLANTA) CHLTYPE(CLUSRCVR)  
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

7. Modifiez les canaux CLUSSDR sur SEATTLE et ATLANTA.

Modifiez les deux définitions de canal CLUSSDR pour ajouter la liste de noms. Dans SEATTLE, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

Dans ATLANTA, entrez la commande suivante:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR)
CLUSTER(' ') CLUSNL(CHAINMAIL)
```

8. Définissez les canaux CLUSRCVR et CLUSSDR sur HARTFORD et OMAHA.

Sur les deux nouveaux gestionnaires de files d'attente HARTFORD et OMAHA, définissez les canaux récepteur de cluster et émetteur de cluster. Peu importe l'ordre dans lequel vous créez les définitions. Dans HARTFORD, entrez:

```
DEFINE CHANNEL(MAILORDER.HARTFORD) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(HARTFORD.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for HARTFORD')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from HARTFORD to repository at CHICAGO')
```

Dans OMAHA, entrez:

```
DEFINE CHANNEL(MAILORDER.OMAHA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(OMAHA.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-receiver channel for OMAHA')

DEFINE CHANNEL(MAILORDER.CHICAGO) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(CHICAGO.CHSTORE.COM) CLUSTER(MAILORDER)
DESCR('Cluster-sender channel from OMAHA to repository at CHICAGO')
```

9. Définissez la file d'attente MORDERQ sur OMAHA.

La dernière étape de cette tâche consiste à définir la file d'attente MORDERQ sur le gestionnaire de files d'attente OMAHA. Dans OMAHA, entrez:

```
DEFINE QLOCAL(MORDERQ) CLUSTER(MAILORDER)
```

10. Vérifiez que les modifications apportées au cluster ont été propagées.

Vérifiez que les définitions que vous avez créées avec les étapes précédentes ont été propagées via le cluster. Exécutez les commandes suivantes sur un gestionnaire de files d'attente de référentiel complet:

```
DIS QCLUSTER (MORDERQ)
DIS CLUSQMGR
```

11.

Résultats

Le cluster configuré par cette tâche est illustré dans la [Figure 46](#), à la page 364.

Nous avons maintenant deux clusters qui se chevauchent. Les référentiels complets des deux clusters se trouvent à l'adresse CHICAGO et CHICAGO2. L'application de commande par courrier qui s'exécute sur OMAHA est indépendante de l'application d'inventaire qui s'exécute sur CHICAGO. Toutefois, certains des gestionnaires de files d'attente qui se trouvent dans le cluster CHNSTORE se trouvent également dans le cluster MAILORDER et peuvent donc envoyer des messages à l'une ou l'autre application. Avant

d'effectuer cette tâche pour chevaucher deux clusters, vous devez être conscient de la possibilité de conflits de noms de file d'attente.

Supposons que sur NEWYORK dans le cluster CHNSTORE et sur OMAHA dans le cluster MAILORDER, il existe une file d'attente appelée ACCOUNTQ. Si vous chevauchez les clusters et qu'une application sur SEATTLE insère un message dans la file d'attente ACCOUNTQ, le message peut être envoyé à l'une des instances de ACCOUNTQ.

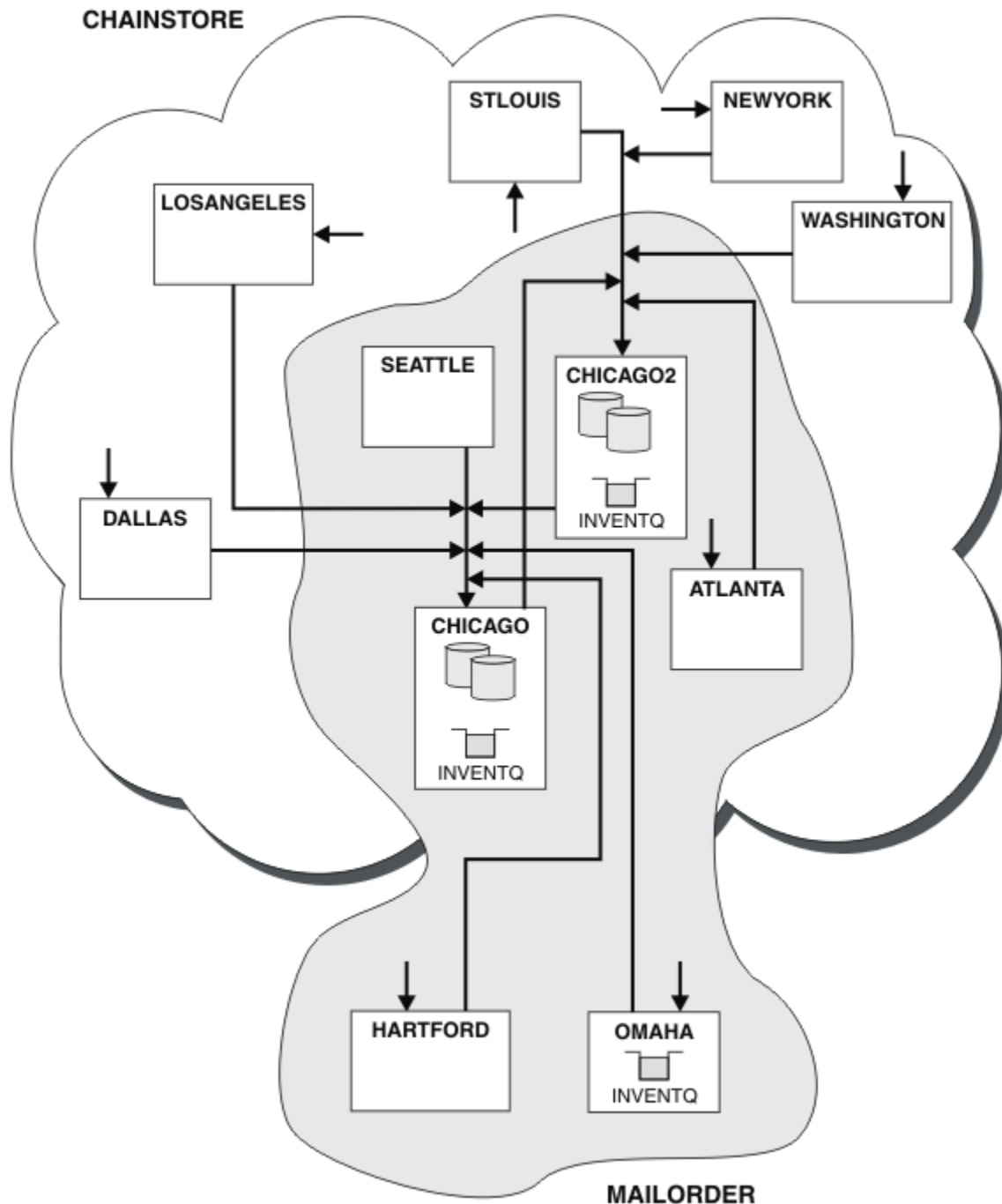


Figure 46. Clusters interconnectés

Que faire ensuite

Supposons que vous décidez de fusionner le cluster MAILORDER avec le cluster CHNSTORE pour former un grand cluster appelé CHNSTORE.

Pour fusionner le cluster MAILORDER avec le cluster CHNSTORE , de sorte que CHICAGO et CHICAGO2 contiennent les référentiels complets:

- Modifiez les définitions de gestionnaire de files d'attente pour CHICAGO et CHICAGO2, en supprimant l'attribut REPOSNL, qui spécifie la liste de noms (CHAINMAIL), et en le remplaçant par un attribut REPOS indiquant le nom du cluster (CHNSTORE). Par exemple:

```
ALTER QMGR(CHICAGO) REPOSNL(' ') REPOS(CHNSTORE)
```

- Sur chaque gestionnaire de files d'attente du cluster MAILORDER , modifiez toutes les définitions de canal et de file d'attente pour modifier la valeur de l'attribut CLUSTER de MAILORDER à CHNSTORE. Par exemple, dans HARTFORD, entrez:

```
ALTER CHANNEL(MAILORDER.HARTFORD) CLUSTER(CHNSTORE)
```

Dans OMAHA , entrez:

```
ALTER QLOCAL(MORDERQ) CLUSTER(CHNSTORE)
```

- Modifiez toutes les définitions qui spécifient la liste de noms de cluster CHAINMAIL, c'est-à-dire les définitions de canal CLUSRCVR et CLUSSDR sur CHICAGO, CHICAGO2, SEATTLEet ATLANTA, pour spécifier à la place le cluster CHNSTORE.

Dans cet exemple, vous pouvez voir l'avantage d'utiliser des listes de noms. Au lieu de modifier les définitions de gestionnaire de files d'attente pour CHICAGO et CHICAGO2 , vous pouvez modifier la valeur de la liste de noms CHAINMAIL. De même, au lieu de modifier les définitions de canal CLUSRCVR et CLUSSDR dans CHICAGO, CHICAGO2, SEATTLEet ATLANTA, vous pouvez obtenir le résultat requis en modifiant la liste de noms.

Tâches associées

Retrait d'un réseau de cluster

Supprimez un cluster d'un réseau et restaurez la configuration de la mise en file d'attente répartie.

Retrait d'un réseau de cluster

Supprimez un cluster d'un réseau et restaurez la configuration de la mise en file d'attente répartie.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Un cluster IBM MQ a été configuré comme décrit dans [«Conversion d'un réseau existant en cluster»](#), à la page 358.
- Ce cluster doit maintenant être supprimé du système. Le réseau des gestionnaires de files d'attente doit continuer à fonctionner comme avant l'implémentation du cluster.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour retirer un réseau de cluster.

Procédure

1. Supprimez les files d'attente de cluster du cluster CHNSTORE .

Sur CHICAGO et CHICAGO2, modifiez la définition de file d'attente locale pour la file d'attente INVENTQ afin de supprimer la file d'attente du cluster. Entrez la commande suivante :

```
ALTER QLOCAL(INVENTQ) CLUSTER(' ')
```

Lorsque vous modifiez la file d'attente, les informations des référentiels complets sont mises à jour et propagées dans le cluster. Les applications actives utilisant MQ00_BIND_NOT_FIXED et les applications utilisant MQ00_BIND_AS_Q_DEF où la file d'attente a été définie avec DEFBIND(NOTFIXED) échouent lors de la prochaine tentative d'appel MQPUT ou MQPUT1 . Le code anomalie MQRC_UNKNOWN_OBJECT_NAME est renvoyé.

Vous n'avez pas à exécuter l'étape 1 en premier, mais si ce n'est pas le cas, vous devez l'exécuter à la place après l'étape 4.

2. Arrêtez toutes les applications ayant accès à la file d'attente de cluster.

Arrêtez toutes les applications ayant accès aux files d'attente de cluster. Si vous ne le faites pas, certaines informations de cluster peuvent rester sur le gestionnaire de files d'attente local lorsque vous actualisez le cluster à l'étape 5. Ces informations sont supprimées lorsque toutes les applications ont été arrêtées et que les canaux de cluster ont été déconnectés.

3. Supprimez l'attribut de référentiel des gestionnaires de files d'attente de référentiel complet.

Sur CHICAGO et CHICAGO2, modifiez les définitions de gestionnaire de files d'attente pour supprimer l'attribut de référentiel. Pour ce faire, exécutez la commande suivante:

```
ALTER QMGR REPOS(' ')
```

Les gestionnaires de files d'attente informent les autres gestionnaires de files d'attente du cluster qu'ils ne contiennent plus les référentiels complets. Lorsque les autres gestionnaires de files d'attente reçoivent ces informations, un message indiquant que le référentiel complet est terminé s'affiche. Vous voyez également un ou plusieurs messages indiquant qu'il n'y a plus de référentiels disponibles pour le cluster CHNSTORE.

4. Supprimez les canaux de cluster.

Sous CHICAGO , retirez les canaux de cluster:

```
ALTER CHANNEL(CHNSTORE.CHICAGO2) CHLTYPE(CLUSSDR) CLUSTER(' ')  
ALTER CHANNEL(CHNSTORE.CHICAGO) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

Remarque : Il est important d'exécuter d'abord la commande CLUSSDR , puis la commande CLUSRCVR . N'émettez pas d'abord la commande CLUSRCVR , puis la commande CLUSSDR . Ce faisant, vous créez des canaux en attente de validation dont le statut est ARRETE . Vous devez ensuite émettre une commande START CHANNEL pour récupérer les canaux arrêtés ; par exemple, START CHANNEL (CHNSTORE . CHICAGO) .

Des messages s'affichent pour indiquer qu'il n'existe aucun référentiel pour le cluster CHNSTORE.

Si vous n'avez pas supprimé les files d'attente de cluster comme décrit à l'étape 1, faites-le maintenant.

5. Arrêtez les canaux de cluster.

Sous CHICAGO , arrêtez les canaux de cluster à l'aide des commandes suivantes:

```
STOP CHANNEL(CHNSTORE.CHICAGO2)  
STOP CHANNEL(CHNSTORE.CHICAGO)
```

6. Répétez les étapes 4 et 5 pour chaque gestionnaire de files d'attente du cluster.

7. Arrêtez les canaux de cluster, puis supprimez toutes les définitions des canaux de cluster et des files d'attente de cluster de chaque gestionnaire de files d'attente.

8. Facultatif : Effacez les informations de cluster en cache détenues par le gestionnaire de files d'attente.

Bien que les gestionnaires de files d'attente ne soient plus membres du cluster, ils conservent chacun une copie en cache des informations sur le cluster. Si vous souhaitez supprimer ces données, voir la tâche «[Restauration d'un gestionnaire de files d'attente à son état antérieur à la mise en cluster](#)», à la [page 397](#).

9. Remplacez les définitions de file d'attente éloignée pour INVENTQ

Pour que le réseau puisse continuer à fonctionner, remplacez la définition de file d'attente éloignée pour INVENTQ sur chaque gestionnaire de files d'attente.

10. Rangez le cluster.

Supprimez les définitions de file d'attente ou de canal qui ne sont plus nécessaires.

Tâches associées

[Ajout d'un nouveau cluster interconnecté](#)

Ajoutez un nouveau cluster qui partage certains gestionnaires de files d'attente avec un cluster existant.

Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle

Suivez les instructions de la tâche pour construire des clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle. Utilisez les clusters comme point de départ pour les exemples suivants qui permettent d'isoler des messages d'une application à partir de messages vers d'autres applications d'un cluster.

Pourquoi et quand exécuter cette tâche

L'exemple de configuration de cluster utilisé pour illustrer l'isolement du trafic des messages de cluster est présenté dans [Figure 47](#), à la [page 368](#). L'exemple est décrit dans [Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster](#).

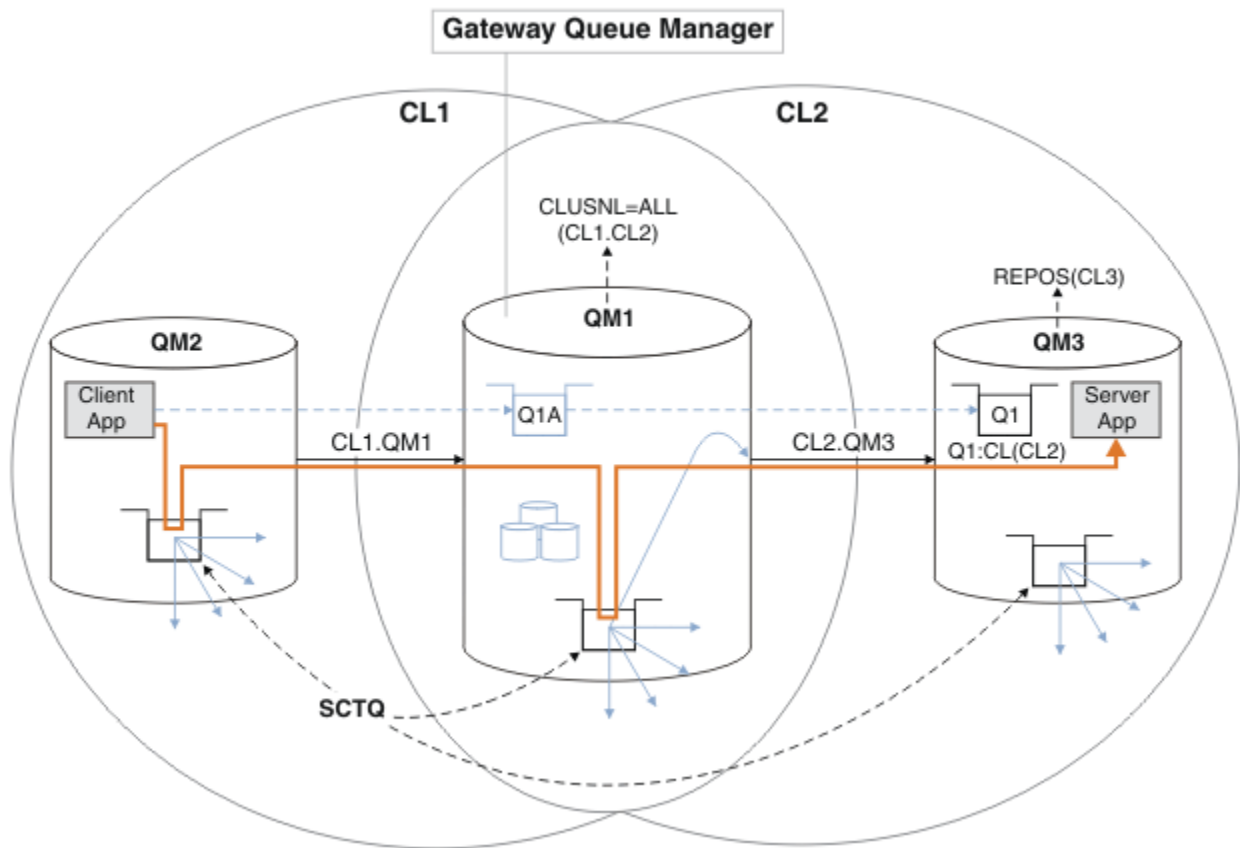


Figure 47. Application client-serveur déployée sur l'architecture concentrateur et satellite à l'aide de clusters IBM MQ

Pour que le nombre d'étapes de construction de l'exemple soit aussi faible que possible, la configuration est plus simple que réaliste. L'exemple peut représenter l'intégration de deux clusters créés par deux organisations distinctes. Pour un scénario plus réaliste, voir [Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster](#).

Procédez comme suit pour construire les clusters. Les clusters sont utilisés dans les exemples suivants pour isoler le trafic de messages de l'application client vers l'application serveur.

Les instructions ajoutent quelques gestionnaires de files d'attente supplémentaires afin que chaque cluster dispose de deux référentiels. Le gestionnaire de files d'attente de passerelle n'est pas utilisé comme référentiel pour des raisons de performances.

Procédure

1. Créez et démarrez les gestionnaires de files d'attente QM1, QM2, QM3, QM4, QM5.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE QM n
strmqm QmgrName
```

Remarque : QM4 et QM5 sont les référentiels complets de sauvegarde pour les clusters.

2. Définissez et démarrez les programmes d'écoute pour chacun des gestionnaires de files d'attente.

```
*... On QM n
DEFINE LISTENER(TCP141 n) TRPTYPE(TCP) IPADDR(hostname) PORT(141 n) CONTROL(QMGR) REPLACE
START LISTENER(TCP141 n)
```

3. Créez une liste de noms de cluster pour tous les clusters.


```
*... On QM1
DEFINE NAMELIST(ALL) NAMES(CL1, CL2) REPLACE
```

4. Créez des référentiels complets QM2 et QM4 pour CL1, QM3 et QM5 pour CL2.

a) Pour CL1 :

```
*... On QM2 and QM4
ALTER QMGR REPOS(CL1) DEFCLXQ(SCTQ)
```

b) Pour CL2 :

```
*... On QM3 and QM5
ALTER QMGR REPOS(CL2) DEFCLXQ(SCTQ)
```

5. Ajoutez les canaux émetteur de cluster et récepteur de cluster pour chaque gestionnaire de files d'attente et cluster.

Exécutez les commandes suivantes sous QM2, QM3, QM4 et QM5, où *c*, *net* *m* prennent les valeurs indiquées dans [Tableau 27](#), à la page 369 pour chaque gestionnaire de files d'attente:

Gestionnaire de files d'attente	Cluster <i>c</i>	Autre référentiel <i>n</i>	Ce référentiel <i>m</i>
QM2	1	4	2
QM4	1	2	4
QM3	2	5	3
QM5	2	3	5

```
*... On QM m
DEFINE CHANNEL(CL c.QM n) CHLTYPE(CLUSSDR) CONNAME('localhost(141 n)') CLUSTER(CL c) REPLACE
DEFINE CHANNEL(CL c.QM m) CHLTYPE(CLUSRCVR) CONNAME('localhost(141 m)') CLUSTER(CL c) REPLACE
```

6. Ajoutez le gestionnaire de files d'attente de passerelle, QM1, à chacun des clusters.

```
*... On QM1
DEFINE CHANNEL(CL1.QM2) CHLTYPE(CLUSSDR) CONNAME('localhost(1412)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL1.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL1) REPLACE
DEFINE CHANNEL(CL2.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL2) REPLACE
DEFINE CHANNEL(CL2.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL2) REPLACE
```

7. Ajoutez la file d'attente locale Q1 au gestionnaire de files d'attente QM3 dans le cluster CL2.

```
*... On QM3
DEFINE QLOCAL(Q1) CLUSTER(CL2) REPLACE
```

8. Ajoutez l'alias de gestionnaire de files d'attente en cluster Q1A au gestionnaire de files d'attente de passerelle.

```
*... On QM1
DEFINE QALIAS(Q1A) CLUSNL(ALL) TARGET(Q1) TARGTYPE(Queue) DEFBIND(NOTFIXED) REPLACE
```

Remarque : Les applications utilisant l'alias de gestionnaire de files d'attente sur tout autre gestionnaire de files d'attente, mais QM1, doivent spécifier DEFBIND(NOTFIXED) lorsqu'elles ouvrent la file d'attente alias. **DEFBIND** indique si les informations de routage de l'en-tête de message sont fixes lorsque la file d'attente est ouverte par l'application. S'il est défini sur la valeur par défaut,

OPEN, les messages sont acheminés vers Q1@QM1. Q1@QM1 n'existe pas, de sorte que les messages provenant d'autres gestionnaires de files d'attente finissent dans une file d'attente de rebut. En définissant l'attribut de file d'attente sur DEFBIND(NOTFIXED), les applications telles que **amqspu**t, qui utilisent par défaut le paramètre de file d'attente **DEFBIND**, se comportent correctement.

9. Ajoutez les définitions d'alias de gestionnaire de files d'attente de cluster pour tous les gestionnaires de files d'attente de cluster au gestionnaire de files d'attente de passerelle, QM1.

```
*... On QM1
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) CLUSNL(ALL) REPLACE
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSNL(ALL) REPLACE
```

Conseil : Les définitions d'alias de gestionnaire de files d'attente sur le gestionnaire de files d'attente de passerelle transfèrent les messages qui font référence à un gestionnaire de files d'attente dans un autre cluster ; voir [Alias de gestionnaire de files d'attente en cluster](#).

Que faire ensuite

1. Testez la définition d'alias de file d'attente en envoyant un message de QM2 à Q1 sur QM3 à l'aide de la définition d'alias de file d'attente Q1A.

- a. Exécutez l'exemple de programme **amqspu**t sur QM2 pour insérer un message.

```
C:\IBM\MQ>amqspu Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

- b. Exécutez l'exemple de programme **amqsge**t pour obtenir le message de Q1 sur QM3

```
C:\IBM\MQ>amqsge Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Testez les définitions d'alias de gestionnaire de files d'attente en envoyant un message de demande et en recevant un message de réponse dans une file d'attente de réponses dynamique temporaire.

Le diagramme montre le chemin emprunté par le message de réponse vers une file d'attente dynamique temporaire, appelée RQ. L'application serveur, connectée à QM3, ouvre la file d'attente de réponses à l'aide du nom de gestionnaire de files d'attente QM2. Le nom du gestionnaire de files d'attente QM2 est défini en tant qu'alias de gestionnaire de files d'attente en cluster sous QM1. QM3 achemine le message de réponse vers QM1. QM1 achemine le message vers QM2.

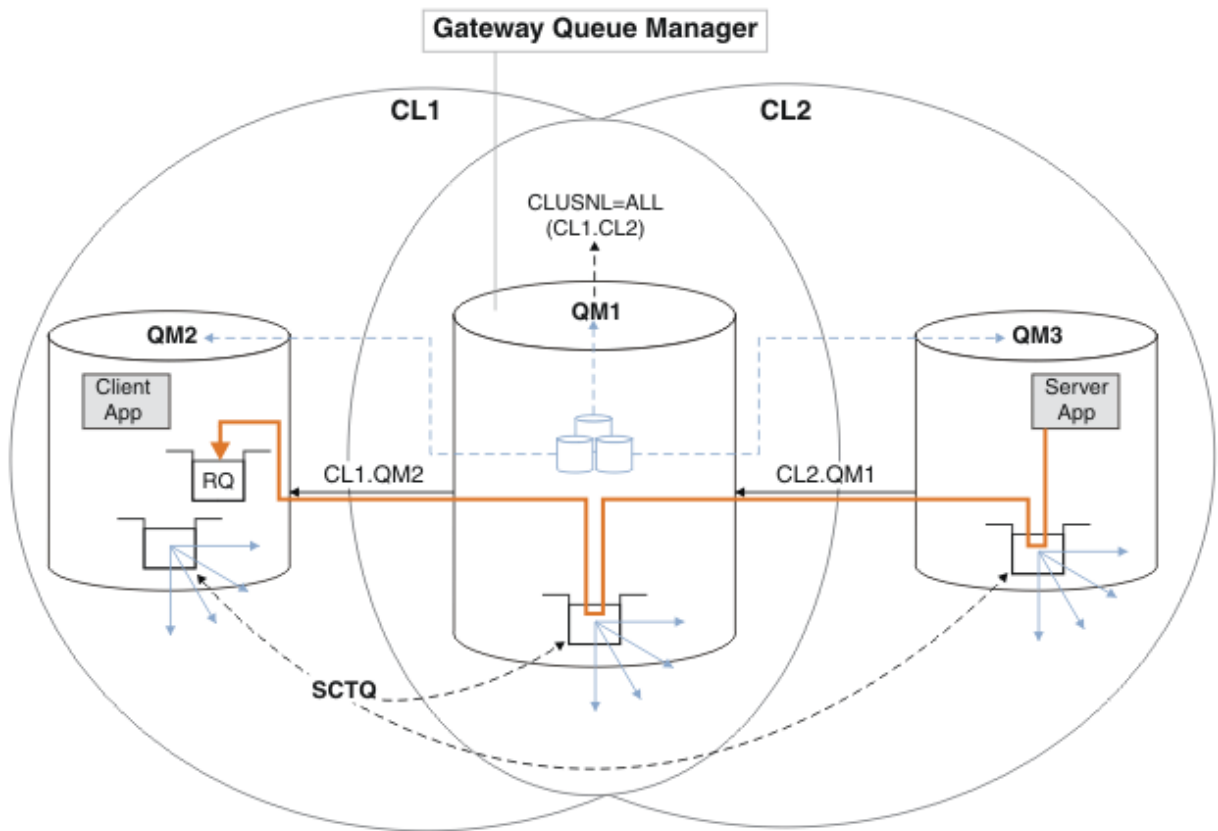


Figure 48. Utilisation d'un alias de gestionnaire de files d'attente pour renvoyer le message de réponse à un autre cluster

Le mode de fonctionnement du routage est le suivant. Chaque gestionnaire de files d'attente de chaque cluster possède une définition d'alias de gestionnaire de files d'attente sous QM1. Les alias sont regroupés dans tous les clusters. Les flèches en pointillés grises de chacun des alias vers un gestionnaire de files d'attente indiquent que chaque alias de gestionnaire de files d'attente est résolu en gestionnaire de files d'attente réel dans au moins un des clusters. Dans ce cas, l'alias QM2 est mis en cluster à la fois dans le cluster CL1 et dans CL2, et est résolu en gestionnaire de files d'attente réel QM2 dans CL1. L'application serveur crée le message de réponse en utilisant le nom de la file d'attente de réponse RQ et le nom du gestionnaire de files d'attente de réponse QM2. Le message est acheminé vers QM1 car la définition d'alias de gestionnaire de files d'attente QM2 est définie sur QM1 dans le cluster CL2 et le gestionnaire de files d'attente QM2 ne se trouve pas dans le cluster CL2. Comme le message ne peut pas être envoyé au gestionnaire de files d'attente cible, il est envoyé au gestionnaire de files d'attente qui possède la définition d'alias.

QM1 place le message dans la file d'attente de transmission du cluster sur QM1 pour le transfert vers QM2. QM1 route le message vers QM2 car la définition d'alias de gestionnaire de files d'attente sous QM1 for QM2 définit QM2 comme le gestionnaire de files d'attente cible réel. La définition n'est pas circulaire, car les définitions d'alias ne peuvent faire référence qu'à des définitions réelles ; l'alias ne peut pas pointer vers lui-même. La définition réelle est résolue par QM1, car QM1 et QM2 se trouvent dans le même cluster, CL1. QM1 recherche les informations de connexion pour QM2 dans le référentiel pour CL1 et achemine le message vers QM2. Pour que le message soit réacheminé par QM1, l'application serveur doit avoir ouvert la file d'attente de réponses avec l'option DEFBIND définie sur MQBND_BIND_NOT_FIXED. Si l'application serveur a ouvert la file d'attente de réponses avec l'option MQBND_BIND_ON_OPEN, le message n'est pas réacheminé et se retrouve dans une file d'attente de messages non livrés.

- a. Créez une file d'attente de demandes en cluster avec un déclencheur sous QM3.

*... On QM3

```
DEFINE QLOCAL(QR) CLUSTER(CL2) TRIGGER INITQ(SYSTEM.DEFAULT.INITIATION.QUEUE)
PROCESS(ECHO) REPLACE
```

- b. Créez une définition d'alias de file d'attente de cluster de QR sur le gestionnaire de files d'attente de passerelle, QM1.

```
*... On QM1
DEFINE QALIAS(QRA) CLUSNL(ALL) TARGET(QR) TARGTYPE(Queue) DEFBIND(NOTFIXED) REPLACE
```

- c. Créez une définition de processus pour démarrer l'exemple de programme echo **amqsech** sur QM3.

```
*... On QM3
DEFINE PROCESS(ECHO) APPLICID(AMQSECH) REPLACE
```

- d. Créez une file d'attente modèle sur QM2 pour l'exemple de programme **amqsreq** afin de créer la file d'attente de réponses dynamique temporaire.

```
*... On QM2
DEFINE QMODEL(SYSTEM.SAMPLE.REPLY) REPLACE
```

- e. Testez la définition d'alias de gestionnaire de files d'attente en envoyant une demande de QM2 à QR sur QM3 à l'aide de la définition d'alias de file d'attente QRA.

- i) Exécutez le programme de moniteur de déclenchement sur QM3.

```
runmqtrm -m QM3
```

La sortie est

```
C:\IBM\MQ>runmqtrm -m QM3
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
01/02/2012 16:17:15: IBM MQ trigger monitor started.
```

```
-----
01/02/2012 16:17:15: Waiting for a trigger message
```

- ii) Exécutez l'exemple de programme **amqsreq** sur QM2 pour placer une demande et attendre une réponse.

```
C:\IBM\MQ>amqsreq QRA QM2
Sample AMQSREQ0 start
server queue is QRA
replies to 4F2961C802290020
A request message from QM2 to QR on QM3
```

```
response <A request message from QM2 to QR on QM3>
no more replies
Sample AMQSREQ0 end
```

Concepts associés

[Contrôle d'accès et files d'attente de transmission de cluster multiples](#)

[Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster](#)

Tâches associées

[Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster](#)

«Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Ajout d'une définition de file d'attente éloignée pour isoler les messages envoyés à partir d'un gestionnaire de files d'attente de passerelle

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une définition éloignée de file d'attente en cluster, ainsi qu'un canal émetteur et une file d'attente de transmission distincts.

Avant de commencer

Construisez les clusters qui se chevauchent dans l'application client-serveur déployée sur l'architecture concentrateur et satellite à l'aide de IBM MQ clusters dans «Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle», à la page 367 en suivant les étapes de cette tâche.

Pourquoi et quand exécuter cette tâche

La solution utilise la mise en file d'attente répartie pour séparer les messages de l'application Server App des autres messages du gestionnaire de files d'attente de passerelle. Vous devez définir une définition de file d'attente éloignée en cluster sur QM1 pour rediriger les messages vers une autre file d'attente de transmission et un autre canal. La définition de file d'attente éloignée doit inclure une référence à la file d'attente de transmission spécifique qui stocke les messages uniquement pour Q1 sur QM3. Dans Figure 49, à la page 374, l'alias de file d'attente de cluster Q1A est complété par une définition de file d'attente éloignée Q1Ret une file d'attente de transmission et un canal émetteur sont ajoutés.

Dans cette solution, tous les messages de réponse sont renvoyés à l'aide du SYSTEM . CLUSTER . TRANSMIT . QUEUEcommun.

L'avantage de cette solution est qu'il est facile de séparer le trafic de plusieurs files d'attente de destination sur le même gestionnaire de files d'attente, dans le même cluster. L'inconvénient de la solution est que vous ne pouvez pas utiliser l'équilibrage de charge de cluster entre plusieurs copies d'Q1 sur des gestionnaires de files d'attente différents. Pour surmonter cet inconvénient, voir «Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 376. Vous devez également gérer le basculement d'une file d'attente de transmission à l'autre.

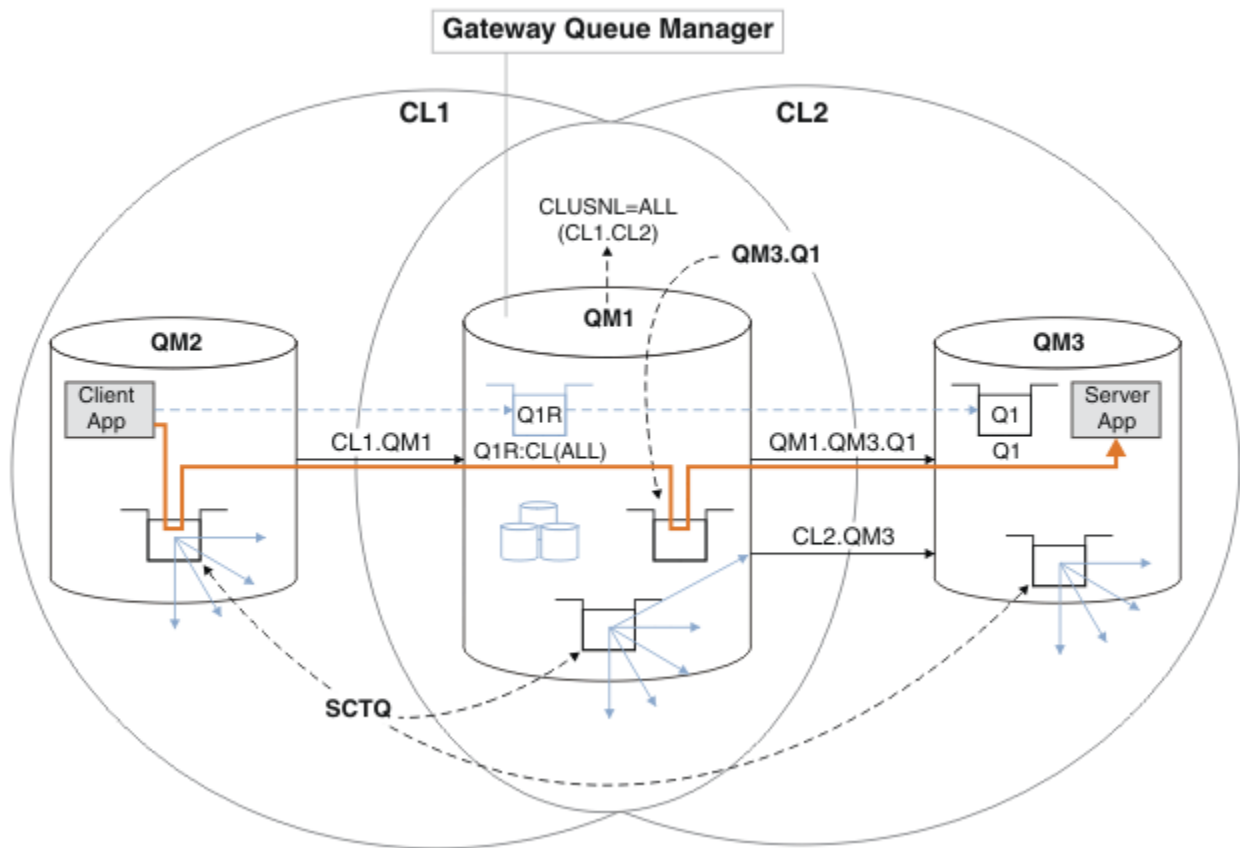


Figure 49. Application client-serveur déployée sur l'architecture de cluster concentrateur et satellite à l'aide de définitions de file d'attente éloignée

Procédure

1. Créez un canal pour séparer le trafic de messages pour Q1 du gestionnaire de files d'attente de passerelle
 - a) Créez un canal émetteur sur le gestionnaire de files d'attente de passerelle, QM1, vers le gestionnaire de files d'attente cible, QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(SDR) CONNAME(QM3HostName(1413)) XMITQ(QM3.Q1) REPLACE
```

- b) Créez un canal récepteur sur le gestionnaire de files d'attente cible, QM3.

```
DEFINE CHANNEL(QM1.QM3.Q1) CHLTYPE(RCVR) REPLACE
```

2. Création d'une file d'attente de transmission sur le gestionnaire de files d'attente de passerelle pour le trafic des messages vers Q1

```
DEFINE QLOCAL(QM3.Q1) USAGE(XMITQ) REPLACE
START CHANNEL(QM1.QM3.Q1)
```

En démarrant le canal associé à la file d'attente de transmission, vous associez la file d'attente de transmission au canal. Le canal démarre automatiquement, une fois que la file d'attente de transmission a été associée au canal.

3. Complétez la définition d'alias de file d'attente en cluster pour Q1 sur le gestionnaire de files d'attente de passerelle avec une définition de file d'attente éloignée en cluster.

```
DEFINE QREMOTE CLUSNL(ALL) RNAME(Q1) RQMNAME(QM3) XMITQ(QM3.Q1) REPLACE
```

Que faire ensuite

Testez la configuration en envoyant un message à Q1 ou QM3 à partir de QM2 à l'aide de la définition éloignée de file d'attente en cluster Q1R sur le gestionnaire de files d'attente de passerelle QM1.

1. Exécutez l'exemple de programme **amqspu**t sur QM2 pour insérer un message.

```
C:\IBM\MQ>amqspu Q1R QM2
Sample AMQSPUT0 start
target queue is Q1R
Sample request message from QM2 to Q1 using Q1R
```

```
Sample AMQSPUT0 end
```

2. Exécutez l'exemple de programme **amqsge**t pour obtenir le message de Q1 sur QM3

```
C:\IBM\MQ>amqsge Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1R>
no more messages
Sample AMQSGET0 end
```

Concepts associés

Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster

Contrôle d'accès et files d'attente de transmission de cluster multiples

Tâches associées

Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une file d'attente de transmission de cluster supplémentaire pour séparer le trafic de messages vers un seul gestionnaire de files d'attente dans un cluster.

Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. , la solution utilise un cluster supplémentaire pour isoler les messages dans une file d'attente de cluster particulière.

Modification de la valeur par défaut pour séparer les files d'attente de transmission de cluster afin d'isoler le trafic des messages

Vous pouvez modifier la façon par défaut dont un gestionnaire de files d'attente stocke les messages d'une file d'attente en cluster ou d'une rubrique dans une file d'attente de transmission. La modification de la valeur par défaut permet d'isoler les messages de cluster sur un gestionnaire de files d'attente de passerelle.

Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster

«Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une file d'attente de transmission de cluster supplémentaire pour séparer le trafic de messages vers un seul gestionnaire de files d'attente dans un cluster.

Avant de commencer

1. Le gestionnaire de files d'attente de passerelle doit être sous IBM MQ.
2. Construisez les clusters qui se chevauchent dans l'application client-serveur déployée sur l'architecture concentrateur et satellite à l'aide de IBM MQ clusters dans «Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle», à la page 367 en suivant les étapes de cette tâche.

Pourquoi et quand exécuter cette tâche

Sur le gestionnaire de files d'attente de passerelle, QM1, ajoutez une file d'attente de transmission et définissez son attribut de file d'attente CLCHNAME. Définissez CLCHNAME sur le nom du canal récepteur de cluster sur QM3 ; voir Figure 50, à la page 377.

Cette solution présente un certain nombre d'avantages par rapport à la solution décrite dans «Ajout d'une définition de file d'attente éloignée pour isoler les messages envoyés à partir d'un gestionnaire de files d'attente de passerelle», à la page 373:

- Il nécessite moins de définitions supplémentaires.
- Il prend en charge l'équilibrage de charge entre plusieurs copies de la file d'attente cible, Q1, sur différents gestionnaires de files d'attente du même cluster, CL2.
- Le gestionnaire de files d'attente de passerelle bascule automatiquement vers la nouvelle configuration lorsque le canal redémarre sans perdre de messages.
- Le gestionnaire de files d'attente de passerelle continue à transmettre les messages dans l'ordre dans lequel il les a reçus. Il le fait, même si le commutateur a lieu avec des messages pour la file d'attente Q1 à QM3 toujours sur SYSTEM . CLUSTER . TRANSMIT . QUEUE.

La configuration permettant d'isoler le trafic des messages de cluster dans Figure 50, à la page 377 ne permet pas d'isoler aussi bien le trafic que la configuration utilisant des files d'attente distantes dans «Ajout d'une définition de file d'attente éloignée pour isoler les messages envoyés à partir d'un gestionnaire de files d'attente de passerelle», à la page 373. Si le gestionnaire de files d'attente QM3 dans CL2 héberge un certain nombre de files d'attente de cluster et d'applications serveur différentes, toutes ces files d'attente partagent le canal de cluster, CL2 . QM3, en connectant QM1 à QM3. Les flux supplémentaires sont illustrés en Figure 50, à la page 377 par la flèche grise représentant le trafic potentiel des messages de cluster entre le SYSTEM . CLUSTER . TRANSMIT . QUEUE et le canal émetteur de cluster CL2 . QM3.

La solution consiste à limiter le gestionnaire de files d'attente à l'hébergement d'une file d'attente de cluster dans un cluster particulier. Si le gestionnaire de files d'attente héberge déjà un certain nombre de files d'attente de cluster, pour respecter cette restriction, vous devez créer un autre gestionnaire de files d'attente ou un autre cluster ; voir «Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 379.

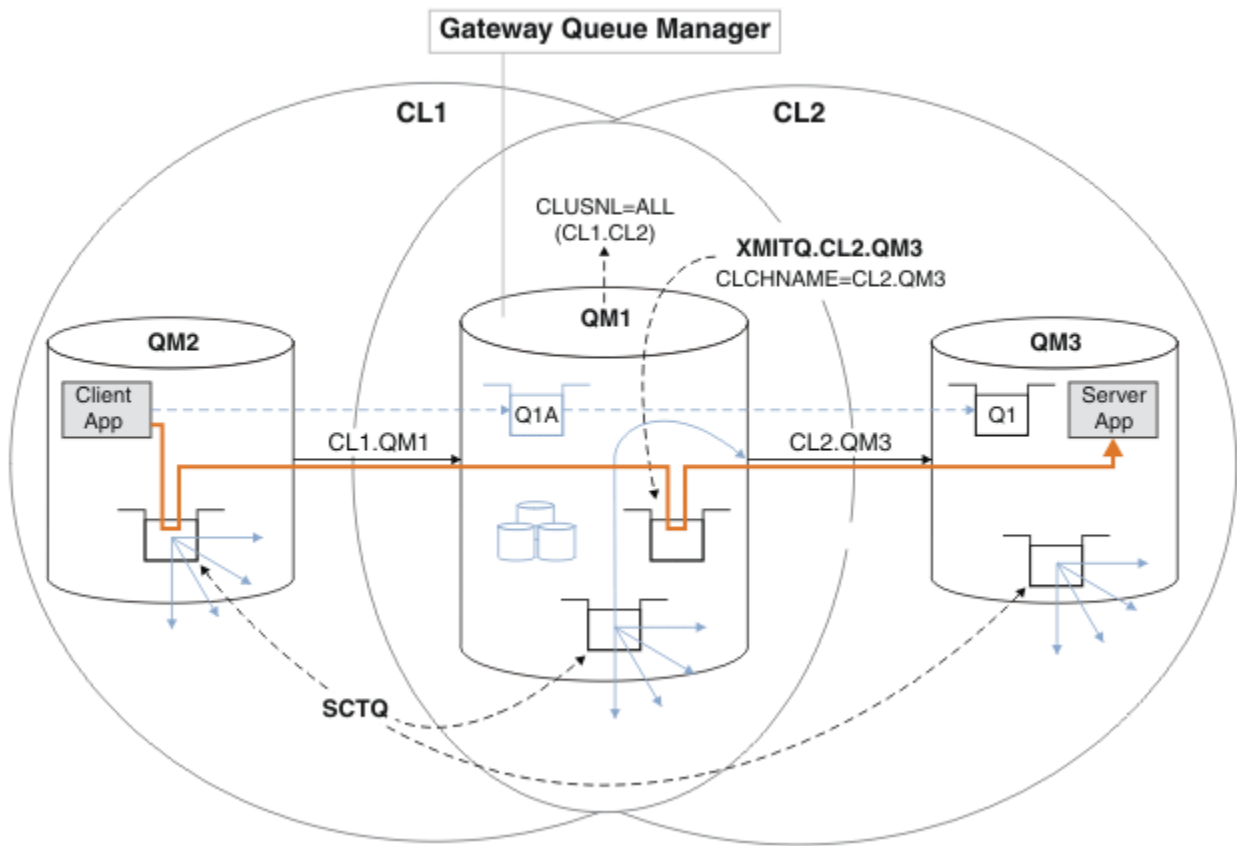


Figure 50. Application client-serveur déployée sur l'architecture concentrateur et satellite à l'aide d'une file d'attente de transmission de cluster supplémentaire.

Procédure

1. Créez une file d'attente de transmission de cluster supplémentaire pour le canal émetteur de cluster CL2.QM3 sur le gestionnaire de files d'attente de passerelle, QM1.

```
*... on QM1
DEFINE QLOCAL(XMITQ.CL2.QM3) USAGE(XMITQ) CLCHNAME(CL2.QM3)
```

2. Passez à l'utilisation de la file d'attente de transmission, XMITQ.CL2.QM3.
 - a) Arrêtez le canal émetteur de cluster CL2.QM3.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

La réponse est que la commande est acceptée:

AMQ8019: Stop IBM MQ channel accepted.

- b) Vérifiez que le canal CL2.QM3 est arrêté

Si le canal ne s'arrête pas, vous pouvez réexécuter la commande **STOP CHANNEL** avec l'option **FORCE**. Par exemple, vous pouvez définir l'option **FORCE** si le canal ne s'arrête pas et que vous ne pouvez pas redémarrer l'autre gestionnaire de files d'attente pour synchroniser le canal.

```
*... On QM1
start
```

La réponse est un récapitulatif du statut du canal

```
AMQ8417: Display Channel Status details.  
CHANNEL (CL2.QM3)           CHLTYPE (CLUSSDR)  
CONNNAME (127.0.0.1(1413))  CURRENT  
RQMNAME (QM3)              STATUS (STOPPED)  
SUBSTATE (MQGET)           XMITQ (SYSTEM.CLUSTER.TRANSMIT.QUEUE)
```

c) Démarrez le canal, CL2.QM3.

```
*... On QM1  
START CHANNEL (CL2.QM3)
```

La réponse est que la commande est acceptée:

```
AMQ8018: Start IBM MQ channel accepted.
```

d) Vérifiez que le canal est démarré.

```
*... On QM1  
DISPLAY CHSTATUS (CL2.QM3)
```

La réponse est un récapitulatif du statut du canal:

```
AMQ8417: Display Channel Status details.  
CHANNEL (CL2.QM3)           CHLTYPE (CLUSSDR)  
CONNNAME (127.0.0.1(1413))  CURRENT  
RQMNAME (QM3)              STATUS (RUNNING)  
SUBSTATE (MQGET)           XMITQ (XMITQ.CL2.QM3)
```

e) Vérifiez que la file d'attente de transmission a été commutée.

Surveillez le journal des erreurs du gestionnaire de files d'attente de passerelle pour le message "AMQ7341 File d'attente de transmission du canal CL2.QM3 est XMITQ.CL2.QM3".

Que faire ensuite

Testez la file d'attente de transmission distincte en envoyant un message de QM2 à Q1 sur QM3 à l'aide de la définition d'alias de file d'attente Q1A

1. Exécutez l'exemple de programme **amqspu**t sur QM2 pour insérer un message.

```
C:\IBM\MQ>amqspu Q1A QM2  
Sample AMQSPUT0 start  
target queue is Q1A  
Sample request message from QM2 to Q1 using Q1A  
  
Sample AMQSPUT0 end
```

2. Exécutez l'exemple de programme **amqsge**t pour obtenir le message de Q1 sur QM3

```
C:\IBM\MQ>amqsge Q1 QM3  
Sample AMQSGET0 start  
message <Sample request message from QM2 to Q1 using Q1A>
```

```
no more messages
Sample AMQSGETO end
```

Concepts associés

[Contrôle d'accès et files d'attente de transmission de cluster multiples](#)

[Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster «Utilisation de files d'attente de transmission de cluster et de canaux émetteurs de cluster», à la page 322](#)

Les messages entre les gestionnaires de files d'attente de cluster sont stockés dans des files d'attente de transmission de cluster et transmis par les canaux émetteurs de cluster. A tout moment, un canal émetteur de cluster est associé à une file d'attente de transmission. Si vous modifiez la configuration du canal, il se peut qu'il passe à une autre file d'attente de transmission lors de son prochain démarrage. Le traitement de ce commutateur est automatisé et transactionnel.

Tâches associées

[Ajout d'une définition de file d'attente éloignée pour isoler les messages envoyés à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une définition éloignée de file d'attente en cluster, ainsi qu'un canal émetteur et une file d'attente de transmission distincts.

[Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. , la solution utilise un cluster supplémentaire pour isoler les messages dans une file d'attente de cluster particulière.

[Modification de la valeur par défaut pour séparer les files d'attente de transmission de cluster afin d'isoler le trafic des messages](#)

Vous pouvez modifier la façon par défaut dont un gestionnaire de files d'attente stocke les messages d'une file d'attente en cluster ou d'une rubrique dans une file d'attente de transmission. La modification de la valeur par défaut permet d'isoler les messages de cluster sur un gestionnaire de files d'attente de passerelle.

[Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster](#)

[«Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345](#)

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. , la solution utilise un cluster supplémentaire pour isoler les messages dans une file d'attente de cluster particulière.

Avant de commencer

Les étapes de la tâche sont écrites pour modifier la configuration illustrée dans la [Figure 50](#), à la page 377.

1. Le gestionnaire de files d'attente de passerelle doit être sous IBM MQ.

2. Construisez les clusters qui se chevauchent dans l'application client-serveur déployée sur l'architecture concentrateur et satellite à l'aide de IBM MQ clusters dans «Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle», à la page 367 en suivant les étapes de cette tâche.
3. Effectuez les étapes décrites dans Figure 50, à la page 377 dans «Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 376 pour créer la solution sans le cluster supplémentaire. Utilisez-la comme base pour les étapes de cette tâche.

Pourquoi et quand exécuter cette tâche

La solution permettant d'isoler le trafic des messages vers une application unique dans «Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 376 fonctionne si la file d'attente de cluster cible est la seule file d'attente de cluster sur un gestionnaire de files d'attente. Si ce n'est pas le cas, vous avez deux choix. Déplacez la file d'attente vers un autre gestionnaire de files d'attente ou créez un cluster qui isole la file d'attente des autres files d'attente de cluster sur le gestionnaire de files d'attente.

Cette tâche vous guide tout au long de la procédure d'ajout d'un cluster pour isoler la file d'attente cible. Le cluster est ajouté uniquement à cette fin. En pratique, vous devez isoler systématiquement certaines applications lorsque vous êtes en train de concevoir des clusters et des schémas de dénomination de cluster. L'ajout d'un cluster chaque fois qu'une file d'attente nécessite un isolement peut se terminer avec de nombreux clusters à gérer. Dans cette tâche, vous modifiez la configuration dans «Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 376 en ajoutant un cluster CL3 pour isoler Q1 sur QM3. Les applications continuent de s'exécuter tout au long de la modification.

Les définitions nouvelles et modifiées sont mises en évidence dans Figure 51, à la page 381. Le récapitulatif des modifications est le suivant: créez un cluster, ce qui signifie que vous devez également créer un nouveau référentiel de cluster complet. Dans l'exemple, QM3 est l'un des référentiels complets pour CL3. Créez des canaux émetteurs et récepteurs de cluster pour QM1 afin d'ajouter le gestionnaire de files d'attente de passerelle au nouveau cluster. Modifiez la définition de Q1 pour la remplacer par CL3. Modifiez la liste de noms de cluster sur le gestionnaire de files d'attente de passerelle et ajoutez une file d'attente de transmission de cluster pour utiliser le nouveau canal de cluster. Enfin, basculez l'alias de file d'attente Q1A vers la nouvelle liste de noms de cluster.

IBM MQ ne peut pas transférer automatiquement les messages de la file d'attente de transmission XMITQ . CL2 . QM3 que vous avez ajoutée dans «Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle», à la page 376 à la nouvelle file d'attente de transmission XMITQ . CL3 . QM3. Il peut transférer des messages automatiquement uniquement si les deux files d'attente de transmission sont servies par le même canal émetteur de cluster. Au lieu de cela, la tâche décrit une façon d'effectuer le commutateur manuellement, ce qui peut être approprié pour vous. Une fois le transfert terminé, vous avez la possibilité de revenir à l'utilisation de la file d'attente de transmission de cluster par défaut pour les autres files d'attente de cluster CL2 sous QM3. Vous pouvez également continuer à utiliser XMITQ . CL2 . QM3. Si vous décidez de revenir à une file d'attente de transmission de cluster par défaut, le gestionnaire de files d'attente de passerelle gère automatiquement le commutateur pour vous.

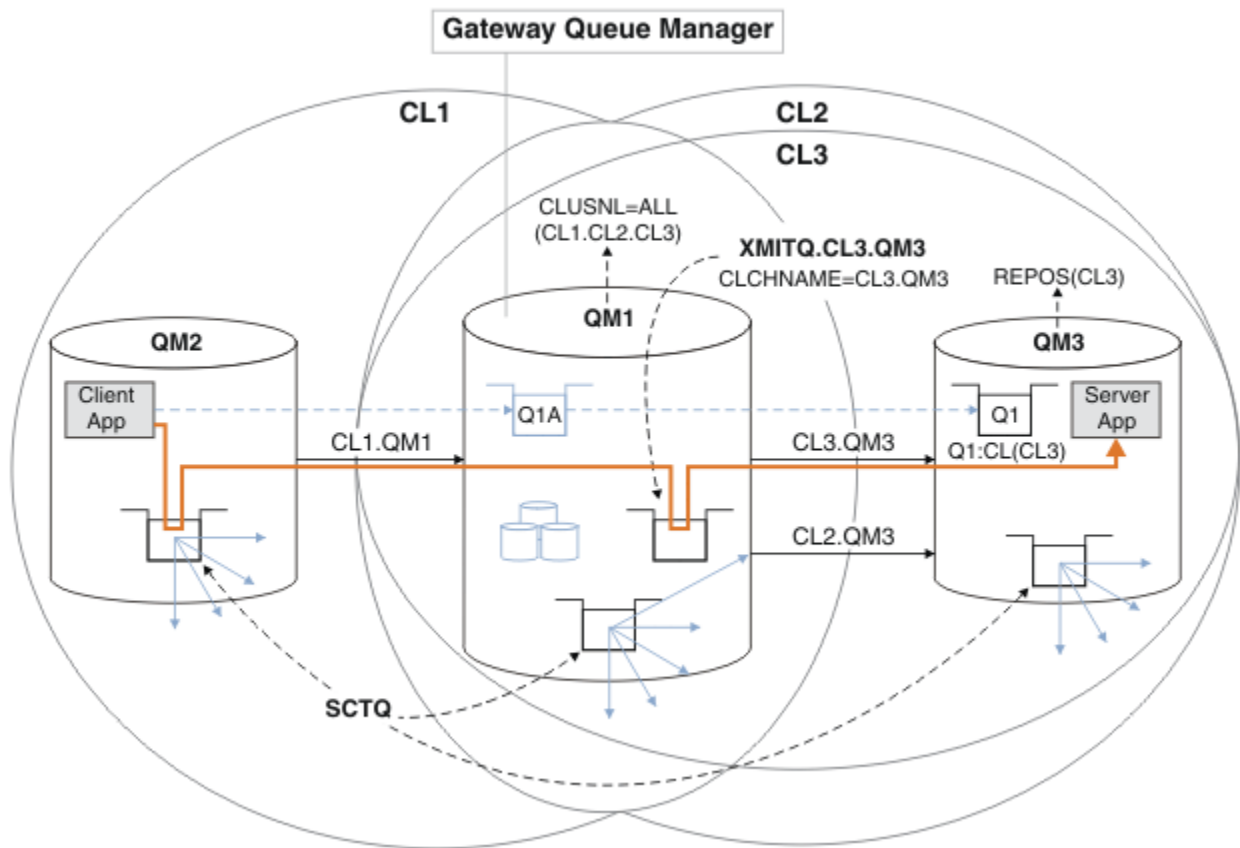


Figure 51. Utilisation d'un cluster supplémentaire pour séparer le trafic de messages dans le gestionnaire de files d'attente de passerelle qui est acheminé vers l'une des nombreuses files d'attente de cluster du même gestionnaire de files d'attente

Procédure

1. Modifiez les gestionnaires de files d'attente QM3 et QM5 pour en faire des référentiels pour CL2 et CL3.

Pour faire d'un gestionnaire de files d'attente un membre de plusieurs clusters, il doit utiliser une liste de noms de cluster pour identifier les clusters dont il est membre.

```
*... On QM3 and QM5
DEFINE NAMELIST(CL23) NAMES(CL2, CL3) REPLACE
ALTER QMGR REPOS(' ') REPOSNL(CL23)
```

2. Définissez les canaux entre les gestionnaires de files d'attente QM3 et QM5 pour CL3.

```
*... On QM3
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSSDR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSRCVR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE

*... On QM5
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
DEFINE CHANNEL(CL3.QM5) CHLTYPE(CLUSRCVR) CONNAME('localhost(1415)') CLUSTER(CL3) REPLACE
```

3. Ajoutez le gestionnaire de files d'attente de passerelle à CL3.

Ajoutez le gestionnaire de files d'attente de passerelle en ajoutant QM1 à CL3 en tant que référentiel partiel. Créez un référentiel partiel en ajoutant des canaux émetteur et récepteur de cluster à QM1.

Ajoutez également CL3 à la liste de noms de tous les clusters connectés au gestionnaire de files d'attente de passerelle.

```
*... On QM1
DEFINE CHANNEL(CL3.QM3) CHLTYPE(CLUSSDR) CONNAME('localhost(1413)') CLUSTER(CL3) REPLACE
```

```
DEFINE CHANNEL(CL3.QM1) CHLTYPE(CLUSRCVR) CONNAME('localhost(1411)') CLUSTER(CL3) REPLACE
ALTER NAMELIST(ALL) NAMES(CL1, CL2, CL3)
```

4. Ajoutez une file d'attente de transmission de cluster au gestionnaire de files d'attente de passerelle, QM1, pour les messages envoyés à CL3 sur QM3.

Au départ, arrêtez le canal émetteur de cluster qui transfère les messages de la file d'attente de transmission jusqu'à ce que vous soyez prêt à basculer entre les files d'attente de transmission.

```
*... On QM1
DEFINE QLOCAL(XMITQ.CL3.QM3) USAGE(XMITQ) CLCHNAME(CL3.QM3) GET(DISABLED) REPLACE
```

5. Purgez les messages de la file d'attente de transmission de cluster existante XMITQ.CL2.QM3.

Cette sous-procédure est destinée à préserver l'ordre des messages dans Q1 pour qu'ils correspondent à l'ordre dans lequel ils sont arrivés au gestionnaire de files d'attente de passerelle. Avec les clusters, l'ordre des messages n'est pas entièrement garanti, mais il est probable qu'il le soit. Si l'ordre des messages est garanti, les applications doivent définir l'ordre des messages ; voir [Ordre dans lequel les messages sont extraits d'une file d'attente](#).

- a) Remplacez la file d'attente cible Q1 sur QM3 CL2 par CL3.

```
*... On QM3
ALTER QLOCAL(Q1) CLUSTER(CL3)
```

- b) Surveillez XMITQ.CL3.QM3 jusqu'à ce que les messages commencent à lui être distribués.

Les messages commencent à être distribués à XMITQ.CL3.QM3 lorsque le basculement de Q1 vers CL3 est propagé au gestionnaire de files d'attente de passerelle.

```
*... On QM1
DISPLAY QUEUE(XMITQ.CL3.QM3) CURDEPTH
```

- c) Surveillez XMITQ.CL2.QM3 jusqu'à ce qu'il n'y ait pas de messages en attente de distribution à Q1 sur QM3.

Remarque : XMITQ.CL2.QM3 peut stocker des messages pour d'autres files d'attente sur QM3 qui sont membres de CL2, auquel cas la profondeur peut ne pas atteindre zéro.

```
*... On QM1
DISPLAY QUEUE(XMITQ.CL2.QM3) CURDEPTH
```

- d) Activez l'obtention à partir de la nouvelle file d'attente de transmission de cluster, XMITQ.CL3.QM3

```
*... On QM1
ALTER QLOCAL(XMITQ.CL3.QM3) GET(ENABLED)
```

6. Supprimez l'ancienne file d'attente de transmission de cluster, XMITQ.CL2.QM3, si elle n'est plus requise.

Les messages des files d'attente de cluster dans CL2 sur QM3 reviennent à l'utilisation de la file d'attente de transmission de cluster par défaut sur le gestionnaire de files d'attente de passerelle, QM1. La file d'attente de transmission de cluster par défaut est SYSTEM.CLUSTER.TRANSMIT.QUEUE ou SYSTEM.CLUSTER.TRANSMIT.CL2.QM3. Celle qui varie selon que la valeur de l'attribut de gestionnaire de files d'attente **DEFCLXQ** sur QM1 est SCTQ ou CHANNEL. Le gestionnaire de files d'attente transfère automatiquement les messages depuis XMITQ.CL2.QM3 lors du prochain démarrage du canal émetteur de cluster CL2.QM3.

- a) Remplacez la file d'attente de transmission, XMITQ.CL2.QM3, qui est une file d'attente de transmission de cluster par une file d'attente de transmission normale.

Cette opération interrompt l'association de la file d'attente de transmission avec les canaux émetteurs de cluster. En réponse, IBM MQ transfère automatiquement les messages de XMITQ.CL2.QM3 vers la file d'attente de transmission de cluster par défaut lors du prochain

démarrage du canal émetteur de cluster. Jusque-là, les messages relatifs à CL2 sur QM3 continuent d'être placés sur XMITQ.CL2.QM3.

```
*... On QM1
ALTER QLOCAL(XMITQ.CL2.QM3) CLCHNAME(' ')
```

b) Arrêtez le canal émetteur de cluster CL2.QM3.

L'arrêt et le redémarrage du canal émetteur de cluster initient le transfert des messages de XMITQ.CL2.QM3 vers la file d'attente de transmission de cluster par défaut. En règle générale, vous devez arrêter et démarrer le canal manuellement pour démarrer le transfert. Le transfert démarre automatiquement si le canal redémarre après l'arrêt à l'expiration de son intervalle de déconnexion.

```
*... On QM1
STOP CHANNEL(CL2.QM3)
```

La réponse est que la commande est acceptée:

```
AMQ8019: Stop IBM MQ channel accepted.
```

c) Vérifiez que le canal CL2.QM3 est arrêté

Si le canal ne s'arrête pas, vous pouvez réexécuter la commande **STOP CHANNEL** avec l'option **FORCE**. Par exemple, vous pouvez définir l'option **FORCE** si le canal ne s'arrête pas et que vous ne pouvez pas redémarrer l'autre gestionnaire de files d'attente pour synchroniser le canal.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

La réponse est un récapitulatif du statut du canal

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))        CURRENT
RQMNAME(QM3)                    STATUS(STOPPED)
SUBSTATE(MQGET)                 XMITQ(XMITQ.CL2.QM3)
```

d) Démarrez le canal, CL2.QM3.

```
*... On QM1
START CHANNEL(CL2.QM3)
```

La réponse est que la commande est acceptée:

```
AMQ8018: Start IBM MQ channel accepted.
```

e) Vérifiez que le canal est démarré.

```
*... On QM1
DISPLAY CHSTATUS(CL2.QM3)
```

La réponse est un récapitulatif du statut du canal:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                CHLTYPE(CLUSSDR)
CONNAME(127.0.0.1(1413))        CURRENT
RQMNAME(QM3)                    STATUS(RUNNING)
SUBSTATE(MQGET)                 XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE/CL2.QM3)
```

- f) Surveillez le journal des erreurs du gestionnaire de files d'attente de passerelle pour le message "AMQ7341 File d'attente de transmission du canal CL2.QM3 est SYSTEM.CLUSTER.TRANSMIT. *QUEUE/CL2.QM3*".
- g) Supprimez la file d'attente de transmission du cluster, XMITQ.CL2.QM3.

```
*... On QM1  
DELETE QLOCAL(XMITQ.CL2.QM3)
```

Que faire ensuite

Testez la file d'attente mise en cluster séparément en envoyant un message de QM2 à Q1 ou QM3 à l'aide de la définition d'alias de file d'attente Q1A

1. Exécutez l'exemple de programme **amqsput** sur QM2 pour insérer un message.

```
C:\IBM\MQ>amqsput Q1A QM2  
Sample AMQSPUT0 start  
target queue is Q1A  
Sample request message from QM2 to Q1 using Q1A
```

```
Sample AMQSPUT0 end
```

2. Exécutez l'exemple de programme **amqsget** pour obtenir le message de Q1 sur QM3

```
C:\IBM\MQ>amqsget Q1 QM3  
Sample AMQSGET0 start  
message <Sample request message from QM2 to Q1 using Q1A>  
no more messages  
Sample AMQSGET0 end
```

Concepts associés

[Contrôle d'accès et files d'attente de transmission de cluster multiples](#)

Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster
[«Utilisation de files d'attente de transmission de cluster et de canaux émetteurs de cluster», à la page 322](#)

Les messages entre les gestionnaires de files d'attente de cluster sont stockés dans des files d'attente de transmission de cluster et transmis par les canaux émetteurs de cluster. A tout moment, un canal émetteur de cluster est associé à une file d'attente de transmission. Si vous modifiez la configuration du canal, il se peut qu'il passe à une autre file d'attente de transmission lors de son prochain démarrage. Le traitement de ce commutateur est automatisé et transactionnel.

Tâches associées

[Ajout d'une définition de file d'attente éloignée pour isoler les messages envoyés à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une définition éloignée de file d'attente en cluster, ainsi qu'un canal émetteur et une file d'attente de transmission distincts.

[Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une file d'attente de transmission de cluster supplémentaire pour séparer le trafic de messages vers un seul gestionnaire de files d'attente dans un cluster.

Modification de la valeur par défaut pour séparer les files d'attente de transmission de cluster afin d'isoler le trafic des messages

Vous pouvez modifier la façon par défaut dont un gestionnaire de files d'attente stocke les messages d'une file d'attente en cluster ou d'une rubrique dans une file d'attente de transmission. La modification de la valeur par défaut permet d'isoler les messages de cluster sur un gestionnaire de files d'attente de passerelle.

Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster

«Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Modification de la valeur par défaut pour séparer les files d'attente de transmission de cluster afin d'isoler le trafic des messages

Vous pouvez modifier la façon par défaut dont un gestionnaire de files d'attente stocke les messages d'une file d'attente en cluster ou d'une rubrique dans une file d'attente de transmission. La modification de la valeur par défaut permet d'isoler les messages de cluster sur un gestionnaire de files d'attente de passerelle.

Avant de commencer

1. Le gestionnaire de files d'attente de passerelle doit être sous IBM MQ.
2. Construisez les clusters qui se chevauchent dans l'application client-serveur déployée sur l'architecture concentrateur et satellite à l'aide de IBM MQ clusters dans «Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle», à la page 367 en suivant les étapes de cette tâche.

Pourquoi et quand exécuter cette tâche

Pour implémenter l'architecture avec plusieurs files d'attente de clusters, votre gestionnaire de files d'attente de passerelle doit se trouver sur IBM MQ. Pour utiliser plusieurs files d'attente de transmission de cluster, vous devez uniquement modifier le type de file d'attente de transmission de cluster par défaut sur le gestionnaire de files d'attente de passerelle. Remplacez la valeur de l'attribut de gestionnaire de files d'attente **DEFCLXQ** on QM1 SCTQ par CHANNEL ; voir Figure 52, à la page 386. Le diagramme montre un flux de messages. Pour les flux vers d'autres gestionnaires de files d'attente ou vers d'autres clusters, le gestionnaire de files d'attente crée des files d'attente de transmission de cluster dynamiques permanentes supplémentaires. Chaque canal émetteur de cluster transfère des messages à partir d'une file d'attente de transmission de cluster différente.

La modification ne prend pas effet immédiatement, sauf si vous connectez le gestionnaire de files d'attente de passerelle aux clusters pour la première fois. La tâche inclut des étapes pour le cas typique de gestion d'une modification apportée à une configuration existante. Pour configurer un gestionnaire de files d'attente afin qu'il utilise des files d'attente de transmission de cluster distinctes lorsqu'il rejoint un cluster pour la première fois, voir «Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345.

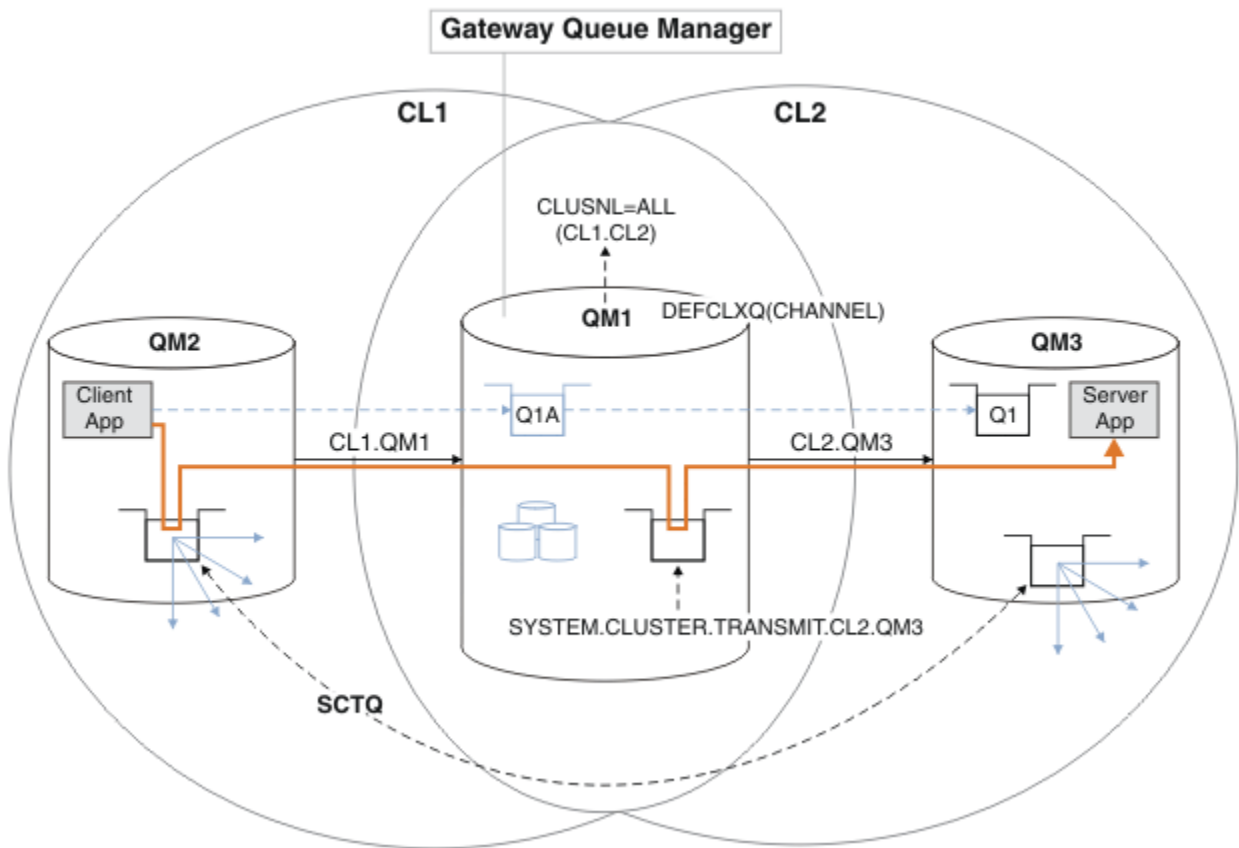


Figure 52. Application client-serveur déployée sur l'architecture concentrateur et satellite avec des files d'attente de transmission de cluster distinctes sur le gestionnaire de files d'attente de passerelle.

Procédure

1. Modifiez le gestionnaire de files d'attente de passerelle pour utiliser des files d'attente de transmission de cluster distinctes.

```
*... On QM1
ALTER QMGR DEFCLXQ(CHANNEL)
```

2. Basculez vers les files d'attente de transmission de cluster distinctes.

Tout canal émetteur de cluster qui n'est pas en cours d'exécution passe à l'utilisation de files d'attente de transmission de cluster distinctes lors du prochain démarrage.

Pour changer les canaux en cours d'exécution, redémarrez le gestionnaire de files d'attente ou procédez comme suit:

- a) Répertoriez les canaux émetteurs de cluster qui s'exécutent avec SYSTEM.CLUSTER.TRANSMIT.QUEUE.

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
```

La réponse est une liste de rapports de statut de canal:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1412))       CURRENT
RQMNAME(QM2)                    STATUS(RUNNING)
```

```

SUBSTATE(MQGET)                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1413))      CURRENT
RQMNAME(QM3)                   STATUS(RUNNING)
SUBSTATE(MQGET)                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1415))      CURRENT
RQMNAME(QM5)                   STATUS(RUNNING)
SUBSTATE(MQGET)                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1414))      CURRENT
RQMNAME(QM4)                   STATUS(RUNNING)
SUBSTATE(MQGET)                XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)

```

b) Arrêter les canaux en cours d'exécution

Pour chaque canal de la liste, exécutez la commande suivante:

```

*... On QM1
STOP CHANNEL(ChannelName)

```

Où *ChannelName* correspond à chacun des CL1.QM2, CL1.QM4, CL1.QM3, CL1.QM5.

La réponse est que la commande est acceptée:

AMQ8019: Stop IBM MQ channel accepted.

c) Surveillance des canaux arrêtés

```

*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')

```

La réponse est une liste de canaux qui sont toujours en cours d'exécution et de canaux qui sont arrêtés:

```

AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1412))      CURRENT
RQMNAME(QM2)                   STATUS(STOPPED)
SUBSTATE( )                     XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1413))      CURRENT
RQMNAME(QM3)                   STATUS(STOPPED)
SUBSTATE( )                     XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1415))      CURRENT
RQMNAME(QM5)                   STATUS(STOPPED)
SUBSTATE( )                     XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1414))      CURRENT
RQMNAME(QM4)                   STATUS(STOPPED)
SUBSTATE( )                     XMITQ(SYSTEM.CLUSTER.TRANSMIT.QUEUE)

```

d) Démarrez chaque canal arrêté.

Effectuez cette étape pour tous les canaux qui étaient en cours d'exécution. Si un canal ne s'arrête pas, vous pouvez exécuter à nouveau la commande **STOP CHANNEL** avec l'option **FORCE**. Par exemple, vous pouvez définir l'option **FORCE** si le canal ne s'arrête pas et que vous ne pouvez pas redémarrer l'autre gestionnaire de files d'attente pour synchroniser le canal.

```
*... On QM1
START CHANNEL(CL2.QM5)
```

La réponse est que la commande est acceptée:

```
AMQ8018: Start IBM MQ channel accepted.
```

e) Surveillez les files d'attente de transmission en cours de commutation.

Surveillez le journal des erreurs du gestionnaire de files d'attente de passerelle pour le message "AMQ7341 File d'attente de transmission du canal CL2.QM3 est SYSTEM.CLUSTER.TRANSMIT. QUEUE/CL2.QM3".

f) Vérifiez que SYSTEM.CLUSTER.TRANSMIT.QUEUE n'est plus utilisé

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ EQ 'SYSTEM.CLUSTER.TRANSMIT.QUEUE')
DISPLAY QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE) CURDEPTH
```

La réponse est une liste de rapports de statut de canal et la profondeur de SYSTEM.CLUSTER.TRANSMIT.QUEUE:

```
AMQ8420: Channel Status not found.
AMQ8409: Display Queue details.
QUEUE(SYSTEM.CLUSTER.TRANSMIT.QUEUE)      TYPE(QLOCAL)
CURDEPTH(0)
```

g) Surveillance des canaux démarrés

```
*... On QM1
DISPLAY CHSTATUS(*) WHERE(XMITQ LK 'SYSTEM.CLUSTER.TRANSMIT.*')
```

La réponse est une liste des canaux, dans ce cas déjà en cours d'exécution avec les nouvelles files d'attente de transmission de cluster par défaut:

```
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM2)                                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1412))                       CURRENT
RQMNAME(QM2)                                     STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL1.QM2)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM3)                                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1413))                       CURRENT
RQMNAME(QM3)                                     STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL2.QM3)
AMQ8417: Display Channel Status details.
CHANNEL(CL2.QM5)                                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1415))                       CURRENT
RQMNAME(QM5)                                     STATUS(RUNNING)
SUBSTATE(MQGET)
```

```
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL2.QM5)
AMQ8417: Display Channel Status details.
CHANNEL(CL1.QM4)                                CHLTYPE(CLUSSDR)
CONNNAME(127.0.0.1(1414))                       CURRENT
RQMNAME(QM4)                                     STATUS(RUNNING)
SUBSTATE(MQGET)
XMITQ(SYSTEM.CLUSTER.TRANSMIT.CL1.QM4)
```

Que faire ensuite

1. Testez la file d'attente de transmission de cluster définie automatiquement en envoyant un message de QM2 à Q1 sur QM3, en résolvant le nom de la file d'attente avec la définition d'alias de file d'attente Q1A

- a. Exécutez l'exemple de programme **amqspu**t sur QM2 pour insérer un message.

```
C:\IBM\MQ>amqspu Q1A QM2
Sample AMQSPUT0 start
target queue is Q1A
Sample request message from QM2 to Q1 using Q1A

Sample AMQSPUT0 end
```

- b. Exécutez l'exemple de programme **amqsge**t pour obtenir le message de Q1 sur QM3

```
C:\IBM\MQ>amqsge Q1 QM3
Sample AMQSGET0 start
message <Sample request message from QM2 to Q1 using Q1A>
no more messages
Sample AMQSGET0 end
```

2. Déterminez s'il convient de reconfigurer la sécurité en configurant la sécurité des files d'attente de cluster sur les gestionnaires de files d'attente d'où proviennent les messages des files d'attente de cluster.

Concepts associés

[Contrôle d'accès et files d'attente de transmission de cluster multiples](#)

[Mise en cluster: isolement d'application à l'aide de plusieurs files d'attente de transmission de cluster](#)

Tâches associées

[Ajout d'une définition de file d'attente éloignée pour isoler les messages envoyés à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une définition éloignée de file d'attente en cluster, ainsi qu'un canal émetteur et une file d'attente de transmission distincts.

[Ajout d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster. La solution utilise une file d'attente de transmission de cluster supplémentaire pour séparer le trafic de messages vers un seul gestionnaire de files d'attente dans un cluster.

[Ajout d'un cluster et d'une file d'attente de transmission de cluster pour isoler le trafic de messages de cluster envoyé à partir d'un gestionnaire de files d'attente de passerelle](#)

Modifiez la configuration des clusters qui se chevauchent et qui utilisent un gestionnaire de files d'attente de passerelle. Une fois les messages de modification transférés à une application à partir du gestionnaire de files d'attente de passerelle, sans utiliser la même file d'attente de transmission ou les mêmes canaux que les autres messages de cluster, la solution utilise un cluster supplémentaire pour isoler les messages dans une file d'attente de cluster particulière.

Mise en cluster: Planification de la configuration des files d'attente de transmission de cluster

«Ajout d'un gestionnaire de files d'attente à un cluster: files d'attente de transmission distinctes», à la page 345

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages envoyés aux files d'attente de cluster et aux rubriques sont transférés à l'aide de plusieurs files d'attente de transmission de cluster.

Suppression d'une file d'attente de cluster d'un gestionnaire de files d'attente

Désactivez la file d'attente INVENTQ à Toronto. Envoyez tous les messages d'inventaire à New York et supprimez la file d'attente INVENTQ à Toronto lorsqu'elle est vide.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans «Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente», à la page 351. Il contient quatre gestionnaires de files d'attente. LONDON et NEWYORK contiennent tous deux des référentiels complets. PARIS et TORONTO contiennent des référentiels partiels. L'application d'inventaire s'exécute sur les systèmes de New York et de Toronto et est gérée par l'arrivée de messages dans la file d'attente INVENTQ .
- En raison d'une charge de travail réduite, vous ne souhaitez plus exécuter l'application d'inventaire à Toronto. Vous souhaitez désactiver la file d'attente INVENTQ hébergée par le gestionnaire de files d'attente TORONTO et placer les messages de flux TORONTO dans la file d'attente INVENTQ dans NEWYORK.
- La connectivité du réseau existe entre les quatre systèmes.
- Le protocole réseau est TCP.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour supprimer une file d'attente de cluster.

Procédure

1. Indique que la file d'attente n'est plus disponible.

Pour supprimer une file d'attente d'un cluster, supprimez le nom du cluster de la définition de file d'attente locale. Modifiez le INVENTQ sur TORONTO de sorte qu'il ne soit pas accessible à partir du reste du cluster:

```
ALTER QLOCAL(INVENTQ) CLUSTER(' ')
```

2. Vérifiez que la file d'attente n'est plus disponible.

Sur un gestionnaire de files d'attente de référentiel complet, LONDON ou NEWYORK, vérifiez que la file d'attente n'est plus hébergée par le gestionnaire de files d'attente TORONTO en exécutant la commande suivante:

```
DIS QCLUSTER (INVENTQ)
```

TORONTO n'est pas répertorié dans les résultats si la commande ALTER a abouti.

3. Désactivez la file d'attente.

Désactivez la file d'attente INVENTQ à l'adresse TORONTO de sorte qu'aucun autre message ne puisse y être écrit:

```
ALTER QLOCAL(INVENTQ) PUT(DISABLED)
```

Désormais, les messages en transit vers cette file d'attente à l'aide de MQOO_BIND_ON_OPEN sont placés dans la file d'attente de rebut. Vous devez empêcher toutes les applications d'insérer explicitement des messages dans la file d'attente de ce gestionnaire de files d'attente.

4. Surveillez la file d'attente jusqu'à ce qu'elle soit vide.

Surveillez la file d'attente à l'aide de la commande DISPLAY QUEUE , en spécifiant les attributs IPPROCS, OPPROCS et CURDEPTH, ou utilisez la commande **WRKMQMSTS** sous IBM i. Lorsque le nombre de processus d'entrée et de sortie et la longueur actuelle des files d'attente sont nuls, la file d'attente est vide.

5. Surveillez le canal pour vous assurer qu'il n'y a pas de messages en attente de validation.

Pour vous assurer qu'il n'y a pas de messages en attente de validation sur le canal INVENTORY.TORONTO, surveillez le canal émetteur de cluster appelé INVENTORY.TORONTO sur chacun des autres gestionnaires de files d'attente. Exécutez la commande DISPLAY CHSTATUS en spécifiant le paramètre INDOUBT à partir de chaque gestionnaire de files d'attente:

```
DISPLAY CHSTATUS(INVENTORY.TORONTO) INDOUBT
```

S'il existe des messages en attente de validation, vous devez les résoudre avant de continuer. Par exemple, vous pouvez essayer d'émettre la commande de canal RESOLVE ou d'arrêter et de redémarrer le canal.

6. Supprimez la file d'attente locale.

Lorsque vous êtes satisfait qu'il n'y a plus de messages à distribuer à l'application d'inventaire à l'adresse TORONTO, vous pouvez supprimer la file d'attente:

```
DELETE QLOCAL(INVENTQ)
```

7. Vous pouvez maintenant supprimer l'application d'inventaire du système à Toronto

Le retrait de l'application permet d'éviter la duplication et d'économiser de l'espace sur le système.

Résultats

Le cluster configuré par cette tâche est similaire à celui configuré par la tâche précédente. La différence est que la file d'attente INVENTQ n'est plus disponible sur le gestionnaire de files d'attente TORONTO.

Lorsque vous avez mis la file d'attente hors service à l'étape 1, le gestionnaire de files d'attente TORONTO a envoyé un message aux deux gestionnaires de files d'attente de référentiel complet. Il les a informés du changement de statut. Les gestionnaires de files d'attente de référentiel complet transmettent ces informations aux autres gestionnaires de files d'attente du cluster qui ont demandé des mises à jour des informations relatives à INVENTQ.

Lorsqu'un gestionnaire de files d'attente insère un message dans la file d'attente INVENTQ , le référentiel partiel mis à jour indique que la file d'attente INVENTQ est disponible uniquement sur le gestionnaire de files d'attente NEWYORK . Le message est envoyé au gestionnaire de files d'attente NEWYORK .

Que faire ensuite

Dans cette tâche, il n'y avait qu'une seule file d'attente à supprimer et un seul cluster à supprimer.

Supposons qu'il existe de nombreuses files d'attente faisant référence à une liste de noms contenant de nombreux noms de cluster. Par exemple, le gestionnaire de files d'attente TORONTO peut héberger non seulement INVENTQ, mais aussi PAYROLLQ, SALESQ et PURCHASESQ. TORONTO rend ces files d'attente disponibles dans tous les clusters appropriés, INVENTORY, PAYROLL, SALES et PURCHASES. Définissez une liste de noms de cluster sur le gestionnaire de files d'attente TORONTO :

```
DEFINE NAMELIST(TOROLIST)
DESCR('List of clusters TORONTO is in')
NAMES(INVENTORY, PAYROLL, SALES, PURCHASES)
```

Ajoutez la liste de noms à chaque définition de file d'attente:

```
DEFINE QLOCAL(INVENTQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PAYROLLQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(SALESQ) CLUSNL(TOROLIST)
DEFINE QLOCAL(PURCHASESQ) CLUSNL(TOROLIST)
```

Supposons maintenant que vous souhaitez supprimer toutes ces files d'attente du cluster SALES , car l'opération SALES doit être reprise par l'opération PURCHASES . Il vous suffit de modifier la liste de noms TOROLIST pour en supprimer le nom du cluster SALES .

Si vous souhaitez supprimer une file d'attente unique de l'un des clusters de la liste de noms, créez une liste de noms contenant la liste restante des noms de cluster. Modifiez ensuite la définition de file d'attente pour utiliser la nouvelle liste de noms. Pour retirer le PAYROLLQ du cluster INVENTORY :

1. Créez une liste de noms:

```
DEFINE NAMELIST(TOROSHORTLIST)
DESCR('List of clusters TORONTO is in other than INVENTORY')
NAMES(PAYROLL, SALES, PURCHASES)
```

2. Modifiez la définition de file d'attente PAYROLLQ :

```
ALTER QLOCAL(PAYROLLQ) CLUSNL(TOROSHORTLIST)
```

Suppression d'un gestionnaire de files d'attente d'un cluster: meilleure pratique

Supprimez un gestionnaire de files d'attente d'un cluster, dans les scénarios où le gestionnaire de files d'attente peut communiquer normalement avec au moins un référentiel complet du cluster.

Avant de commencer

Cette méthode est recommandée pour les scénarios dans lesquels au moins un référentiel complet est disponible et peut être contacté par le gestionnaire de files d'attente en cours de suppression. Cette méthode implique la moindre intervention manuelle et permet au gestionnaire de files d'attente de négocier un retrait contrôlé du cluster. Si le gestionnaire de files d'attente en cours de suppression ne peut pas contacter un référentiel complet, voir [«Suppression d'un gestionnaire de files d'attente d'un cluster: méthode alternative»](#), à la page 395.

Pourquoi et quand exécuter cette tâche

Cet exemple de tâche supprime le gestionnaire de files d'attente LONDON du cluster INVENTORY . Le cluster INVENTORY est configuré comme décrit dans [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343 et modifié comme décrit dans [«Suppression d'une file d'attente de cluster d'un gestionnaire de files d'attente»](#), à la page 390.

Le processus de suppression d'un gestionnaire de files d'attente d'un cluster est plus compliqué que le processus d'ajout d'un gestionnaire de files d'attente.

Lorsqu'un gestionnaire de files d'attente rejoint un cluster, les membres existants du cluster ne connaissent pas le nouveau gestionnaire de files d'attente et n'ont donc aucune interaction avec ce dernier. De nouveaux canaux émetteur et récepteur doivent être créés sur le gestionnaire de files d'attente de jointure pour qu'il puisse se connecter à un référentiel complet.

Lorsqu'un gestionnaire de files d'attente est supprimé d'un cluster, il est probable que les applications connectées au gestionnaire de files d'attente utilisent des objets tels que des files d'attente hébergées ailleurs dans le cluster. En outre, les applications qui sont connectées à d'autres gestionnaires de files d'attente dans le cluster peuvent utiliser des objets hébergés sur le gestionnaire de files d'attente cible. Grâce à ces applications, le gestionnaire de files d'attente en cours peut créer des canaux émetteurs supplémentaires pour établir la communication avec des membres de cluster autres que le référentiel complet qu'il a utilisé pour rejoindre le cluster. Chaque gestionnaire de files d'attente du cluster possède une copie en cache des données qui décrit les autres membres du cluster. Cela peut inclure celui qui est en cours de suppression.

Procédure

1. Avant de supprimer le gestionnaire de files d'attente du cluster, assurez-vous qu'il n'héberge plus les ressources requises par le cluster:
 - Si le gestionnaire de files d'attente héberge un référentiel complet, effectuez les étapes 1 à 6 de la section [«Déplacement d'un référentiel complet vers un autre gestionnaire de files d'attente»](#), à la page 355. Si la fonctionnalité de référentiel complet du gestionnaire de files d'attente à supprimer ne doit pas être déplacée vers un autre gestionnaire de files d'attente, il suffit d'exécuter les étapes 5 et 6.
 - Si le gestionnaire de files d'attente héberge des files d'attente de cluster, effectuez les étapes 1 à 7 de la section [«Suppression d'une file d'attente de cluster d'un gestionnaire de files d'attente»](#), à la page 390.
 - Si le gestionnaire de files d'attente héberge des rubriques de cluster, supprimez-les (par exemple, à l'aide de la commande `DELETE TOPIC`) ou déplacez-les vers d'autres hôtes, comme décrit dans [«Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente»](#), à la page 476.

Remarque : Si vous supprimez un gestionnaire de files d'attente d'un cluster et que le gestionnaire de files d'attente héberge toujours une rubrique de cluster, le gestionnaire de files d'attente peut continuer à tenter de distribuer des publications aux gestionnaires de files d'attente qui restent dans le cluster jusqu'à ce que la rubrique soit supprimée.

2. Modifiez les canaux récepteurs de cluster définis manuellement pour les supprimer du cluster, sur le gestionnaire de files d'attente LONDON:

```
ALTER CHANNEL(INVENTORY.LONDON) CHLTYPE(CLUSRCVR) CLUSTER(' ')
```

3. Modifiez les canaux émetteurs de cluster définis manuellement pour les supprimer du cluster, sur le gestionnaire de files d'attente LONDON:

```
ALTER CHANNEL(INVENTORY.PARIS) CHLTYPE(CLUSSDR) CLUSTER(' ')
```

Les autres gestionnaires de files d'attente du cluster apprennent que ce gestionnaire de files d'attente et ses ressources de cluster ne font plus partie du cluster.

4. Surveillez la file d'attente de transmission du cluster, sur le gestionnaire de files d'attente LONDON, jusqu'à ce qu'il n'y ait aucun message en attente de flux vers un référentiel complet du cluster.

```
DISPLAY CHSTATUS(INVENTORY.PARIS) XQMSGSA
```

Si les messages restent dans la file d'attente de transmission, déterminez la raison pour laquelle ils ne sont pas envoyés aux référentiels complets PARIS et NEWYORK avant de continuer.

Résultats

Le gestionnaire de files d'attente LONDON ne fait plus partie du cluster. Cependant, il peut toujours fonctionner en tant que gestionnaire de files d'attente indépendant.

Que faire ensuite

Le résultat de ces modifications peut être confirmé en exécutant la commande suivante sur les membres restants du cluster:

```
DISPLAY CLUSQMGR(LONDON)
```

Le gestionnaire de files d'attente continue de s'afficher jusqu'à ce que les canaux émetteurs de cluster définis automatiquement soient arrêtés. Vous pouvez attendre que cela se produise ou continuer à surveiller les instances actives en exécutant la commande suivante:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Lorsque vous êtes certain qu'aucun autre message n'est distribué à ce gestionnaire de files d'attente, vous pouvez arrêter les canaux émetteurs de cluster vers LONDON en exécutant la commande suivante sur les membres restants du cluster:

```
STOP CHANNEL(INVENTORY.LONDON) STATUS(INACTIVE)
```

Une fois que les modifications ont été propagées dans le cluster et qu'aucun autre message n'a été distribué à ce gestionnaire de files d'attente, arrêtez et supprimez le canal CLUSRCVR sur LONDON:

```
STOP CHANNEL(INVENTORY.LONDON)  
DELETE CHANNEL(INVENTORY.LONDON)
```

Si une file d'attente de transmission définie manuellement était utilisée pour ce canal et que le modèle CLCHNAME ne correspond à aucun autre canal existant ou planifié, vous pouvez supprimer la file d'attente de transmission. Exemple :

```
DELETE QLOCAL(PARIS.CUSTOM.XMITQ)
```

Remarque : S'il s'agit de files d'attente de transmission définies automatiquement ou du système SYSTEM.CLUSTER.TRANSMIT.QUEUE sont en cours d'utilisation, cette étape n'est pas requise.

Le gestionnaire de files d'attente supprimé peut être ajouté au cluster ultérieurement, comme décrit dans [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343. Le gestionnaire de files d'attente supprimé continue de mettre en cache les connaissances des membres restants du cluster pendant 90 jours maximum. Si vous préférez ne pas attendre l'expiration de ce cache, vous pouvez le supprimer de force comme décrit dans [«Restauration d'un gestionnaire de files d'attente à son état antérieur à la mise en cluster»](#), à la page 397.

Tâches associées

[Suppression d'un gestionnaire de files d'attente d'un cluster \(à l'aide de IBM MQ Explorer\)](#)

Référence associée

[ALTER CHANNEL \(modification des paramètres de canal\)](#)

[DISPLAY CHANNEL \(définition de canal d'affichage\)](#)

[DISPLAY CHSTATUS \(statut du canal d'affichage\)](#)

[DISPLAY CLUSQMGR \(affichage des informations de canal pour les gestionnaires de files d'attente de cluster\)](#)

[STOP CHANNEL \(arrêt d'un canal\)](#)

Suppression d'un gestionnaire de files d'attente d'un cluster: méthode alternative

Supprimez un gestionnaire de files d'attente d'un cluster, dans les scénarios où, en raison d'un problème de système ou de configuration important, le gestionnaire de files d'attente ne peut pas communiquer avec un référentiel complet du cluster.

Avant de commencer

Cette autre méthode de suppression manuelle d'un gestionnaire de files d'attente d'un cluster permet d'arrêter et de supprimer tous les canaux de cluster qui relient le gestionnaire de files d'attente supprimé au cluster et de le supprimer de manière forcée du cluster. Cette méthode est utilisée dans les scénarios où le gestionnaire de files d'attente en cours de suppression ne peut communiquer avec aucun des référentiels complets. Cela peut être dû, par exemple, à l'arrêt du fonctionnement du gestionnaire de files d'attente ou à un incident de communication prolongé entre le gestionnaire de files d'attente et le cluster. Sinon, utilisez la méthode la plus courante: [«Suppression d'un gestionnaire de files d'attente d'un cluster: meilleure pratique»](#), à la page 392.

Pourquoi et quand exécuter cette tâche

Cet exemple de tâche supprime le gestionnaire de files d'attente LONDON du cluster INVENTORY . Le cluster INVENTORY est configuré comme décrit dans [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343 et modifié comme décrit dans [«Suppression d'une file d'attente de cluster d'un gestionnaire de files d'attente»](#), à la page 390.

Le processus de suppression d'un gestionnaire de files d'attente d'un cluster est plus compliqué que le processus d'ajout d'un gestionnaire de files d'attente.

Lorsqu'un gestionnaire de files d'attente rejoint un cluster, les membres existants du cluster ne connaissent pas le nouveau gestionnaire de files d'attente et n'ont donc aucune interaction avec ce dernier. De nouveaux canaux émetteur et récepteur doivent être créés sur le gestionnaire de files d'attente de jointure pour qu'il puisse se connecter à un référentiel complet.

Lorsqu'un gestionnaire de files d'attente est supprimé d'un cluster, il est probable que les applications connectées au gestionnaire de files d'attente utilisent des objets tels que des files d'attente hébergées ailleurs dans le cluster. En outre, les applications qui sont connectées à d'autres gestionnaires de files d'attente dans le cluster peuvent utiliser des objets hébergés sur le gestionnaire de files d'attente cible. Grâce à ces applications, le gestionnaire de files d'attente en cours peut créer des canaux émetteurs supplémentaires pour établir la communication avec des membres de cluster autres que le référentiel complet qu'il a utilisé pour rejoindre le cluster. Chaque gestionnaire de files d'attente du cluster possède une copie en cache des données qui décrit les autres membres du cluster. Cela peut inclure celui qui est en cours de suppression.

Cette procédure peut être appropriée en cas d'urgence, lorsqu'il n'est pas possible d'attendre que le gestionnaire de files d'attente quitte correctement le cluster.

Procédure

1. Avant de supprimer le gestionnaire de files d'attente du cluster, assurez-vous qu'il n'héberge plus les ressources requises par le cluster:

- Si le gestionnaire de files d'attente héberge un référentiel complet, effectuez les étapes 1 à 6 de la section «Déplacement d'un référentiel complet vers un autre gestionnaire de files d'attente», à la page 355. Si la fonctionnalité de référentiel complet du gestionnaire de files d'attente à supprimer ne doit pas être déplacée vers un autre gestionnaire de files d'attente, il suffit d'exécuter les étapes 5 et 6.
- Si le gestionnaire de files d'attente héberge des files d'attente de cluster, effectuez les étapes 1 à 7 de la section «Suppression d'une file d'attente de cluster d'un gestionnaire de files d'attente», à la page 390.
- Si le gestionnaire de files d'attente héberge des rubriques de cluster, supprimez-les (par exemple, à l'aide de la commande DELETE TOPIC) ou déplacez-les vers d'autres hôtes, comme décrit dans «Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente», à la page 476.

Remarque : Si vous supprimez un gestionnaire de files d'attente d'un cluster et que le gestionnaire de files d'attente héberge toujours une rubrique de cluster, le gestionnaire de files d'attente peut continuer à tenter de distribuer des publications aux gestionnaires de files d'attente qui restent dans le cluster jusqu'à ce que la rubrique soit supprimée.

2. Arrêtez tous les canaux utilisés pour communiquer avec les autres gestionnaires de files d'attente du cluster. Utilisez MODE (FORCE) pour arrêter le canal CLUSRCVR sur le gestionnaire de files d'attente LONDON. Sinon, vous devrez peut-être attendre que le gestionnaire de files d'attente émetteur arrête le canal:

```
STOP CHANNEL (INVENTORY.LONDON) MODE(FORCE)
STOP CHANNEL (INVENTORY.TORONTO)
STOP CHANNEL (INVENTORY.PARIS)
STOP CHANNEL (INVENTORY.NEWYORK)
```

3. Surveillez les états des canaux, sur le gestionnaire de files d'attente LONDON, jusqu'à ce que les canaux s'arrêtent:

```
DISPLAY CHSTATUS (INVENTORY.LONDON)
DISPLAY CHSTATUS (INVENTORY.TORONTO)
DISPLAY CHSTATUS (INVENTORY.PARIS)
DISPLAY CHSTATUS (INVENTORY.NEWYORK)
```

Aucun autre message d'application n'est envoyé vers ou depuis les autres gestionnaires de files d'attente du cluster après l'arrêt des canaux.

4. Supprimez les canaux de cluster définis manuellement sur le gestionnaire de files d'attente LONDON:

```
DELETE CHANNEL (INVENTORY.NEWYORK)
DELETE CHANNEL (INVENTORY.TORONTO)
```

5. Les autres gestionnaires de files d'attente du cluster conservent la connaissance du gestionnaire de files d'attente supprimé et peuvent continuer à lui envoyer des messages. Pour purger les connaissances des gestionnaires de files d'attente restants, réinitialisez le gestionnaire de files d'attente supprimé du cluster sur l'un des référentiels complets:

```
RESET CLUSTER(INVENTORY) ACTION(FORCEREMOVE) QMNAME(LONDON) QUEUES(YES)
```

S'il se peut qu'un autre gestionnaire de files d'attente du cluster porte le même nom que le gestionnaire de files d'attente supprimé, indiquez la valeur **QMID** du gestionnaire de files d'attente supprimé.

Résultats

Le gestionnaire de files d'attente LONDON ne fait plus partie du cluster. Cependant, il peut toujours fonctionner en tant que gestionnaire de files d'attente indépendant.

Que faire ensuite

Le résultat de ces modifications peut être confirmé en exécutant la commande suivante sur les membres restants du cluster:

```
DISPLAY CLUSQMGR(LONDON)
```

Le gestionnaire de files d'attente continue de s'afficher jusqu'à ce que les canaux émetteurs de cluster définis automatiquement soient arrêtés. Vous pouvez attendre que cela se produise ou continuer à surveiller les instances actives en exécutant la commande suivante:

```
DISPLAY CHANNEL(INVENTORY.LONDON)
```

Une fois que les modifications sont propagées dans le cluster et qu'aucun autre message n'est distribué à ce gestionnaire de files d'attente, supprimez le canal CLUSRCVR sur LONDON:

```
DELETE CHANNEL(INVENTORY.LONDON)
```

Le gestionnaire de files d'attente supprimé peut être ajouté au cluster ultérieurement, comme décrit dans «Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343. Le gestionnaire de files d'attente supprimé continue de mettre en cache les connaissances des membres restants du cluster pendant 90 jours maximum. Si vous préférez ne pas attendre l'expiration de ce cache, vous pouvez le supprimer de force comme décrit dans «Restauration d'un gestionnaire de files d'attente à son état antérieur à la mise en cluster», à la page 397.

Référence associée

[DELETE CHANNEL \(suppression d'un canal\)](#)

[DISPLAY CHANNEL \(définition de canal d'affichage\)](#)

[DISPLAY CHSTATUS \(statut du canal d'affichage\)](#)

[DISPLAY CLUSQMGR \(affichage des informations de canal pour les gestionnaires de files d'attente de cluster\)](#)

[STOP CHANNEL \(arrêt d'un canal\)](#)

[RESET CLUSTER \(réinitialisation d'un cluster\)](#)

Restauration d'un gestionnaire de files d'attente à son état antérieur à la mise en cluster

Lorsqu'un gestionnaire de files d'attente est supprimé d'un cluster, il conserve la connaissance des autres membres du cluster. Ces connaissances arrivent à expiration et sont supprimées automatiquement. Toutefois, si vous préférez le supprimer immédiatement, vous pouvez utiliser les étapes de cette rubrique.

Avant de commencer

Il est supposé que le gestionnaire de files d'attente a été supprimé du cluster et qu'il n'effectue plus aucun travail dans le cluster. Par exemple, ses files d'attente ne reçoivent plus de messages du cluster et aucune application n'attend de messages dans ces files d'attente.

Pourquoi et quand exécuter cette tâche

Lorsqu'un gestionnaire de files d'attente est supprimé d'un cluster, il conserve la connaissance des autres membres du cluster pendant 90 jours au maximum. Cela peut présenter des avantages pour le système, en particulier si le gestionnaire de files d'attente rejoint rapidement le cluster. Lorsque cette connaissance arrive à expiration, elle est supprimée automatiquement. Toutefois, il existe des raisons pour lesquelles vous pouvez préférer supprimer ces informations manuellement. Exemple :

- Vous pouvez confirmer que vous avez arrêté toutes les applications de ce gestionnaire de files d'attente qui utilisaient auparavant des ressources de cluster. Jusqu'à l'expiration de la connaissance des

membres de cluster restants, toute application de ce type continue à écrire dans une file d'attente de transmission. Une fois la connaissance du cluster supprimée, le système génère un message d'erreur lorsqu'une application de ce type tente d'utiliser les ressources du cluster.

- Lorsque vous affichez les informations de statut du gestionnaire de files d'attente, vous pouvez préférer ne pas voir les informations d'expiration sur les membres de cluster restants.

Cette tâche utilise le cluster INVENTORY comme exemple. Le gestionnaire de files d'attente LONDON a été supprimé du cluster INVENTORY, comme décrit dans [«Suppression d'un gestionnaire de files d'attente d'un cluster: meilleure pratique»](#), à la page 392. Pour supprimer la connaissance des autres membres du cluster, exécutez les commandes suivantes sur le gestionnaire de files d'attente LONDON.

Procédure

1. Supprimez toute la mémoire des autres gestionnaires de files d'attente du cluster de ce gestionnaire de files d'attente:

```
REFRESH CLUSTER(INVENTORY) REPOS(YES)
```

2. Surveillez le gestionnaire de files d'attente jusqu'à ce que toutes les ressources de cluster soient utilisées:

```
DISPLAY CLUSQMR(*) CLUSTER(INVENTORY)  
DISPLAY QCLUSTER(*) CLUSTER(INVENTORY)  
DISPLAY TOPIC(*) CLUSTER(INVENTORY)
```

Concepts associés

[Clusters](#)

[Composants de cluster](#)

Référence associée

[Comparaison de la mise en cluster et de la mise en file d'attente répartie](#)

Gestion d'un gestionnaire de files d'attente

Interrompez et reprenez un gestionnaire de files d'attente à partir d'un cluster pour effectuer la maintenance.

Pourquoi et quand exécuter cette tâche

De temps à autre, il peut être nécessaire d'effectuer des opérations de maintenance sur un gestionnaire de files d'attente faisant partie d'un cluster. Par exemple, vous pouvez être amené à effectuer des sauvegardes des données dans ses files d'attente ou à appliquer des correctifs au logiciel. Si le gestionnaire de files d'attente héberge des files d'attente, ses activités doivent être interrompues. Une fois la maintenance terminée, ses activités peuvent être reprises.

Procédure

1. Interrompez un gestionnaire de files d'attente à l'aide de la commande `SUSPEND QMGR runmqsc` :

```
SUSPEND QMGR CLUSTER(SALES)
```

La commande `SUSPEND runmqsc` avertit les gestionnaires de files d'attente du cluster SALES que ce gestionnaire de files d'attente a été suspendu.

L'objectif de la commande `SUSPEND QMGR` est uniquement de conseiller aux autres gestionnaires de files d'attente d'éviter d'envoyer des messages à ce gestionnaire de files d'attente si possible. Cela ne signifie pas que le gestionnaire de files d'attente est désactivé. Certains messages qui doivent être traités par ce gestionnaire de files d'attente lui sont toujours envoyés, par exemple lorsque ce gestionnaire de files d'attente est le seul hôte d'une file d'attente en cluster.

Lorsque le gestionnaire de files d'attente est interrompu, les routines de gestion de charge de travail évitent de lui envoyer des messages. Les messages qui doivent être traités par ce gestionnaire de files d'attente incluent les messages envoyés par le gestionnaire de files d'attente local.

IBM MQ utilise un algorithme d'équilibrage de charge pour déterminer les destinations qui conviennent, plutôt que de sélectionner le gestionnaire de files d'attente local dans la mesure du possible.

- a) Appliquez la mise en suspens d'un gestionnaire de files d'attente à l'aide de l'option FORCE sur la commande SUSPEND QMGR :

```
SUSPEND QMGR CLUSTER(SALES) MODE(FORCE)
```

MODE (FORCE) force l'arrêt de tous les canaux entrants des autres gestionnaires de files d'attente du cluster. Si vous ne spécifiez pas MODE (FORCE), la valeur par défaut MODE (QUIESCE) s'applique.


2. Effectuez les tâches de maintenance nécessaires.
3. Reprenez le gestionnaire de files d'attente en exécutant la commande RESUME QMGR **runmqsc** :

```
RESUME QMGR CLUSTER(SALES)
```


Résultats

La commande RESUME **runmqsc** indique aux référentiels complets que le gestionnaire de files d'attente est à nouveau disponible. Les gestionnaires de files d'attente de référentiel complet diffusent ces informations à d'autres gestionnaires de files d'attente qui ont demandé des mises à jour des informations concernant ce gestionnaire de files d'attente.

Gestion de la file d'attente de transmission du cluster

Faites tout ce qui est en votre pouvoir pour que les files d'attente de transmission du cluster restent disponibles. Elles sont essentielles aux performances des clusters.  Sous z/OS, définissez INDXTYPE d'une file d'attente de transmission de cluster sur CORRELID.

Avant de commencer

- Assurez-vous que la file d'attente de transmission du cluster n'est pas saturée.
- Prenez soin de ne pas émettre de commande ALTER **runmqsc** pour la désactiver ou la désactiver accidentellement.
- Assurez-vous que le support sur lequel la file d'attente de transmission du cluster est stockée dans  (par exemple, z/OS ensembles de pages) n'est pas saturé.

Pourquoi et quand exécuter cette tâche



La procédure suivante s'applique uniquement à z/OS.

Procédure

Définissez INDXTYPE de la file d'attente de transmission de cluster sur CORRELID

Régénération d'un gestionnaire de files d'attente de cluster

Vous pouvez supprimer des canaux définis automatiquement et des objets de cluster définis automatiquement du référentiel local à l'aide de la commande REFRESH CLUSTER . Aucun message n'est perdu.

Avant de commencer

Il se peut que vous soyez invité à utiliser la commande par votre centre de support IBM . N'utilisez pas la commande sans une attention particulière. Par exemple, pour les grands clusters, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster pendant qu'il est en cours, et de nouveau à des intervalles de 27 jours par la suite lorsque les objets de cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir [Mise en cluster: utilisation des meilleures pratiques REFRESH CLUSTER](#).

Pourquoi et quand exécuter cette tâche

Un gestionnaire de files d'attente peut effectuer un nouveau démarrage dans un cluster. Dans des circonstances normales, vous n'avez pas besoin d'utiliser la commande **REFRESH CLUSTER** .

Procédure

Exécutez la commande **REFRESH CLUSTER MQSC** à partir d'un gestionnaire de files d'attente pour supprimer le gestionnaire de files d'attente de cluster défini automatiquement et les objets de file d'attente du référentiel local.

La commande supprime uniquement les objets qui font référence à d'autres gestionnaires de files d'attente, elle ne supprime pas les objets liés au gestionnaire de files d'attente local. La commande supprime également les canaux définis automatiquement. Elle supprime les canaux qui ne comportent pas de messages dans la file d'attente de transmission du cluster et qui ne sont pas connectés à un gestionnaire de files d'attente de référentiel complet.

Résultats

En effet, la commande **REFRESH CLUSTER** permet à un gestionnaire de files d'attente d'être démarré à froid en ce qui concerne son contenu de référentiel complet. IBM MQ garantit qu'aucune donnée n'est perdue dans vos files d'attente.

Information associée

[Mise en cluster : meilleures pratiques d'utilisation REFRESH CLUSTER](#)

Récupération d'un gestionnaire de files d'attente de cluster

Mettez à jour les informations de cluster relatives à un gestionnaire de files d'attente à l'aide de la commande **REFRESH CLUSTER runmqsc** . Suivez cette procédure après avoir récupéré un gestionnaire de files d'attente à partir d'une sauvegarde ponctuelle.

Avant de commencer

Vous avez restauré un gestionnaire de files d'attente de cluster à partir d'une sauvegarde ponctuelle.

Pourquoi et quand exécuter cette tâche

Pour récupérer un gestionnaire de files d'attente dans un cluster, restaurez le gestionnaire de files d'attente, puis mettez à jour les informations du cluster à l'aide de la commande **REFRESH CLUSTER runmqsc** .

Remarque : Pour les clusters de grande taille, l'utilisation de la commande **REFRESH CLUSTER** peut perturber le cluster lors de son exécution, ainsi qu'ensuite, à des intervalles de 27 jours, lorsque les objets du cluster envoient automatiquement des mises à jour de statut à tous les gestionnaires de files d'attente intéressés. Voir [L'actualisation d'un grand cluster peut affecter les performances et la disponibilité du cluster](#).

Procédure

Exécutez la commande **REFRESH CLUSTER** sur le gestionnaire de files d'attente restauré pour tous les clusters auxquels le gestionnaire de files d'attente participe.

Que faire ensuite

Il n'est pas nécessaire d'exécuter la commande REFRESH CLUSTER sur un autre gestionnaire de files d'attente.

Concepts associés

[Mise en cluster : meilleures pratiques d'utilisation REFRESH CLUSTER](#)

Configuration des canaux de cluster pour la disponibilité

Suivez les bonnes pratiques de configuration pour assurer le bon fonctionnement des canaux de cluster en cas d'arrêts intermittents du réseau.

Avant de commencer

Les clusters vous dispensent de la nécessité de définir des canaux, mais vous devez toujours les gérer. La même technologie de canal est utilisée pour la communication entre les gestionnaires de files d'attente d'un cluster que celle utilisée dans la mise en file d'attente répartie. Pour comprendre les canaux de cluster, vous devez être familiarisé avec des sujets tels que:

- Fonctionnement des canaux
- Comment trouver leur statut
- Utilisation des exits de canal

Pourquoi et quand exécuter cette tâche

Vous souhaitez peut-être accorder une attention particulière aux points suivants:

Procédure

Tenez compte des points suivants lors de la configuration des canaux de cluster

- Choisissez des valeurs pour HBINT ou Kaint sur les canaux émetteurs de cluster et les canaux récepteurs de cluster qui ne constituent pas une charge pour le réseau avec de nombreux flux de signal de présence ou de signal de présence. Un intervalle inférieur à environ 10 secondes donne de faux échecs, si votre réseau ralentit parfois et introduit des délais de cette longueur.
- Définissez la valeur BATCHHB afin de réduire la fenêtre pour provoquer un message abandonné car il est en attente de validation sur un canal défaillant. Un lot en attente de validation sur un canal défaillant est plus susceptible de se produire si le lot est plus long à remplir. Si le trafic des messages sur le canal est sporadique avec de longues périodes entre les rafales de messages, un lot ayant échoué est plus probable.
- Un problème se produit si l'extrémité émettrice de cluster d'un canal échoue, puis tente de redémarrer avant que le signal de présence ou le signal de présence n'ait détecté l'échec. Le redémarrage de l'émetteur de canal est rejeté si l'extrémité réceptrice de cluster du canal est restée active. Pour éviter l'échec, faites en sorte que le canal récepteur de cluster soit arrêté et redémarré lorsqu'un canal émetteur de cluster tente de redémarrer.

Activé IBM MQ for z/OS

Contrôlez le problème de l'extrémité réceptrice de cluster du canal restant actif à l'aide des paramètres **ADOPTMCA** et **ADOPTCHK** sur [ALTER QMGR](#).

Activé Multiplateformes

Contrôlez le problème de l'extrémité réceptrice de cluster du canal restant actif à l'aide des attributs **AdoptNewMCA**, **AdoptNewMCATimeout** et **AdoptNewMCACheck** du fichier [qm.ini](#) ou du registre Windows .

Exemple

Pour obtenir des exemples d'implémentation de ces paramètres sur toutes les plateformes, voir [«Paramètres suggérés», à la page 254](#) .

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 *routage direct* dans un cluster, vérifiez que l'abonnement de proxy attendu existe sur les autres référentiels partiels du cluster. Si vous utilisez le *routage 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 *routage direct*) ou sur les autres hôtes de rubrique (si vous utilisez le *routage 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.

Référence associée

 [Comportement asynchrone des commandes CLUSTER sous z/OS](#)

Routage des messages vers et depuis des clusters

Utilisez des alias de file d'attente, des alias de gestionnaire de files d'attente et des définitions de file d'attente éloignée pour connecter des clusters à des gestionnaires de files d'attente externes et à d'autres clusters.

Pour plus de détails sur le routage des messages vers et depuis les clusters, voir les sous-rubriques suivantes:

Concepts associés

[Clusters](#)

[Composants d'un cluster](#)

[«Alias de gestionnaire de files d'attente et clusters», à la page 418](#)

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

[«Alias de file d'attente et clusters», à la page 422](#)

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

[«Alias de file d'attente de réponse et clusters», à la page 422](#)

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Tâches associées

[«Configuration d'un cluster de gestionnaires de files d'attente», à la page 316](#)

Les clusters fournissent un mécanisme d'interconnexion des gestionnaires de files d'attente qui simplifie à la fois la configuration initiale et la gestion en cours. Vous pouvez définir des composants de cluster et créer et gérer des clusters.

«[Configuration d'un nouveau cluster](#)», à la page 331

Suivez ces instructions pour configurer l'exemple de cluster. Des instructions distinctes décrivent la configuration du cluster sur TCP/IP, LU 6.2 et avec une ou plusieurs files d'attente de transmission. Testez le cluster en envoyant un message d'un gestionnaire de files d'attente à l'autre.

Référence associée

[Comparaison de la mise en cluster et de la mise en file d'attente répartie](#)

Configuration de la demande / réponse à un cluster

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

Avant de commencer

La [Figure 53](#), à la page 406 montre un gestionnaire de files d'attente appelé QM3 qui se trouve en dehors du cluster appelé DEMO. QM3 peut être un gestionnaire de files d'attente sur un produit IBM MQ qui ne prend pas en charge les clusters. QM3 héberge une file d'attente appelée Q3, qui est définie comme suit:

```
DEFINE QLOCAL(Q3)
```

Dans le cluster se trouvent deux gestionnaires de files d'attente appelés QM1 et QM2. QM2 héberge une file d'attente de cluster appelée Q2, qui est définie comme suit:

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO)
```

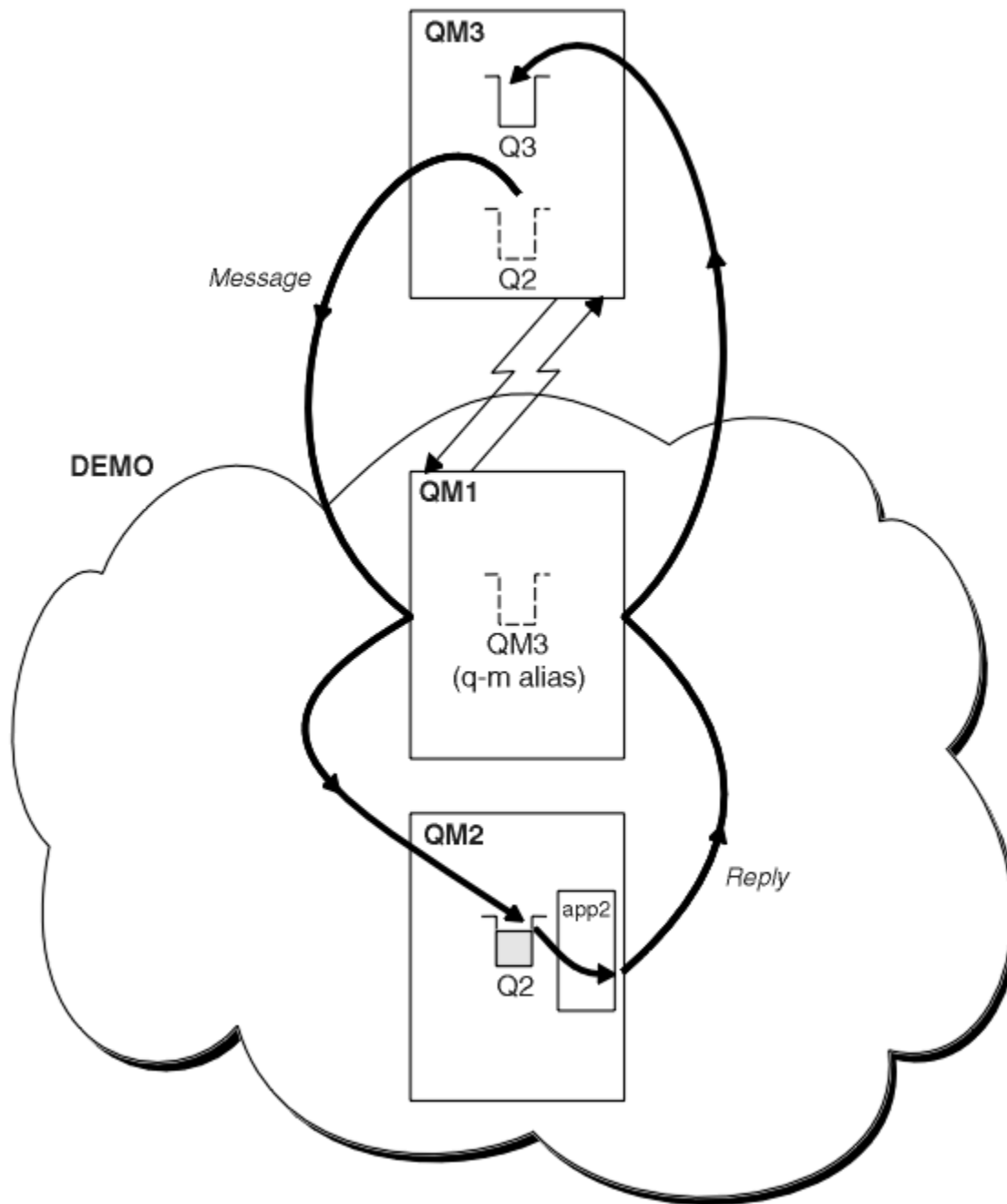


Figure 53. Insertion à partir d'un gestionnaire de files d'attente en dehors du cluster

Pourquoi et quand exécuter cette tâche

Suivez les conseils de la procédure pour définir le chemin des messages de demande et de réponse.

Procédure

1. Envoyez le message de demande au cluster.

Tenez compte de la façon dont le gestionnaire de files d'attente qui se trouve en dehors du cluster insère un message dans la file d'attente Q2 à l'adresse QM2, c'est-à-dire à l'intérieur du cluster. Un gestionnaire de files d'attente en dehors du cluster doit disposer d'une définition QREMOTE pour chaque file d'attente du cluster dans laquelle il insère des messages.

- a) Définissez une file d'attente éloignée pour Q2 sur QM3.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(QM2) XMITQ(QM1)
```

Etant donné que QM3 ne fait pas partie d'un cluster, il doit communiquer à l'aide de techniques de mise en file d'attente répartie. Par conséquent, il doit également disposer d'un canal émetteur et d'une file d'attente de transmission vers QM1. QM1 a besoin d'un canal récepteur correspondant. Les canaux et les files d'attente de transmission ne sont pas affichés explicitement dans [Figure 53](#), à la [page 406](#).

Dans l'exemple, une application à l'adresse QM3 émet un appel MQPUT pour insérer un message dans Q2. La définition QREMOTE entraîne le routage du message vers Q2 à l'adresse QM2 à l'aide du canal émetteur qui extrait les messages de la file d'attente de transmission QM1 .

2. Recevez le message de réponse du cluster.

Utilisez un alias de gestionnaire de files d'attente pour créer un chemin de retour pour les réponses à un gestionnaire de files d'attente en dehors du cluster. La passerelle, QM1, annonce un alias de gestionnaire de files d'attente pour le gestionnaire de files d'attente situé en dehors du cluster, QM3. Il annonce QM3 aux gestionnaires de files d'attente du cluster en ajoutant l'attribut de cluster à une définition d'alias de gestionnaire de files d'attente pour QM3. Une définition d'alias de gestionnaire de files d'attente est similaire à une définition de file d'attente éloignée, mais avec un RNAMEvide.

a) Définissez un alias de gestionnaire de files d'attente pour QM3 sur QM1.

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

Nous devons prendre en compte le choix du nom de la file d'attente de transmission utilisée pour réacheminer les réponses de QM1 vers QM3. Implicitement dans la définition QREMOTE , par omission de l'attribut XMITQ , le nom de la file d'attente de transmission est QM3. Mais QM3 est le même nom que celui que nous prévoyons d'annoncer au reste du cluster à l'aide de l'alias de gestionnaire de files d'attente. IBM MQ ne vous permet pas d'attribuer le même nom à la file d'attente de transmission et à l'alias du gestionnaire de files d'attente. Une solution consiste à créer une file d'attente de transmission pour transmettre des messages à QM3 avec un nom différent à l'alias du gestionnaire de files d'attente.

b) Indiquez le nom de la file d'attente de transmission dans la définition QREMOTE .

```
DEFINE QREMOTE(QM3) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO) XMITQ(QM3.XMIT)
```

Le nouvel alias de gestionnaire de files d'attente couple la nouvelle file d'attente de transmission appelée QM3 .XMIT avec l'alias de gestionnaire de files d'attente QM3 . Il s'agit d'une solution simple et correcte, mais pas entièrement satisfaisante. Il a enfreint la convention de dénomination des files d'attente de transmission en leur attribuant le même nom que le gestionnaire de files d'attente cible. Existe-t-il d'autres solutions qui préservent la convention de dénomination des files d'attente de transmission?

Le problème se produit car le demandeur a transmis par défaut QM3 comme nom de gestionnaire de files d'attente de réponse dans le message de demande envoyé par QM3. Le serveur sous QM2 utilise le nom du gestionnaire de files d'attente de réponse QM3 pour l'adresse QM3 dans ses réponses. La solution a requis QM1 pour annoncer QM3 comme alias de gestionnaire de files d'attente pour renvoyer les messages de réponse à et a empêché QM1 d'utiliser QM3 comme nom de la file d'attente de transmission.

Au lieu de fournir par défaut QM3 comme nom de gestionnaire de files d'attente de réponse, les applications sous QM3 doivent transmettre un alias de gestionnaire de files d'attente de réponse à QM1 pour les messages de réponse. Le gestionnaire de files d'attente de passerelle QM1 annonce l'alias du gestionnaire de files d'attente pour les réponses à QM3 plutôt qu'à QM3 lui-même, ce qui évite le conflit avec le nom de la file d'attente de transmission.

c) Définissez un alias de gestionnaire de files d'attente pour QM3 sur QM1.

```
DEFINE QREMOTE(QM3.ALIAS) RNAME(' ') RQMNAME(QM3) CLUSTER(DEMO)
```

Deux modifications doivent être apportées aux commandes de configuration.

- i) L'QREMOTE à l'adresse QM1 annonce désormais l'alias de gestionnaire de files d'attente QM3 . ALIAS au reste du cluster, en le couplant au nom du gestionnaire de files d'attente réel QM3. QM3 est à nouveau le nom de la file d'attente de transmission pour renvoyer les files d'attente de réponses à QM3
- ii) L'application client doit fournir QM3 . ALIAS comme nom du gestionnaire de files d'attente de réponse lorsqu'elle construit le message de demande. Vous pouvez fournir QM3 . ALIAS à l'application client de l'une des deux manières suivantes.
 - Code QM3 . ALIAS dans la zone du nom du gestionnaire de files d'attente de réponse construite par MQPUT dans MQMD. Vous devez le faire de cette manière si vous utilisez une file d'attente dynamique pour les réponses.
 - Utilisez un alias de file d'attente de réponse, Q3 . ALIAS, plutôt qu'une file d'attente de réponse lorsque vous indiquez le nom de la file d'attente de réponse.

```
DEFINE QREMOTE(Q3.ALIAS) RNAME(Q3) RQMNAME(QM3.ALIAS)
```

Que faire ensuite

Remarque : Vous ne pouvez pas démontrer l'utilisation d'alias de file d'attente de réponse avec **AMQSREQ0**. Elle ouvre la file d'attente de réponse en utilisant le nom de file d'attente fourni dans le paramètre 3 ou la file d'attente modèle SYSTEM . SAMPLE . REPLY par défaut. Vous devez modifier l'exemple en fournissant un autre paramètre contenant l'alias de file d'attente de réponse pour nommer l'alias de gestionnaire de files d'attente de réponse pour MQPUT.

Concepts associés

Alias de gestionnaire de files d'attente et clusters

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

Alias de file d'attente de réponse et clusters

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Alias de file d'attente et clusters

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Tâches associées

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

Configuration des chemins d'accès aux messages entre les clusters

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

«Masquage du nom d'un gestionnaire de files d'attente cible de cluster», à la page 409

Acheminez un message vers une file d'attente de cluster définie sur n'importe quel gestionnaire de files d'attente d'un cluster sans nommer le gestionnaire de files d'attente.

Masquage du nom d'un gestionnaire de files d'attente cible de cluster

Acheminez un message vers une file d'attente de cluster définie sur n'importe quel gestionnaire de files d'attente d'un cluster sans nommer le gestionnaire de files d'attente.

Avant de commencer

- Evitez de révéler les noms des gestionnaires de files d'attente qui se trouvent dans le cluster aux gestionnaires de files d'attente qui se trouvent en dehors du cluster.
 - La résolution des références au gestionnaire de files d'attente hébergeant une file d'attente dans le cluster supprime la souplesse d'équilibrage de la charge de travail.
 - Il est également difficile pour vous de modifier un gestionnaire de files d'attente hébergeant une file d'attente dans le cluster.
 - Vous pouvez également remplacer RQMNAME par un alias de gestionnaire de files d'attente fourni par l'administrateur de cluster.
 - [«Masquage du nom d'un gestionnaire de files d'attente cible de cluster»](#), à la page 409 décrit l'utilisation d'un alias de gestionnaire de files d'attente pour découpler un gestionnaire de files d'attente en dehors d'un cluster de la gestion des gestionnaires de files d'attente dans un cluster.
- Toutefois, la méthode recommandée pour nommer les files d'attente de transmission consiste à leur attribuer le nom du gestionnaire de files d'attente cible. Le nom de la file d'attente de transmission indique le nom d'un gestionnaire de files d'attente dans le cluster. Vous devez choisir la règle à suivre. Vous pouvez choisir de nommer la file d'attente de transmission à l'aide du nom du gestionnaire de files d'attente ou du nom du cluster:

Nommez la file d'attente de transmission à l'aide du nom du gestionnaire de files d'attente de passerelle

La divulgation du nom du gestionnaire de files d'attente de passerelle aux gestionnaires de files d'attente en dehors d'un cluster constitue une exception raisonnable à la règle de masquage des noms de gestionnaire de files d'attente de cluster.

Nommez la file d'attente de transmission à l'aide du nom du cluster

Si vous ne respectez pas la convention de dénomination des files d'attente de transmission avec le nom du gestionnaire de files d'attente cible, utilisez le nom du cluster.

Pourquoi et quand exécuter cette tâche

Modifiez la tâche [«Configuration de la demande / réponse à un cluster»](#), à la page 405 pour masquer le nom du gestionnaire de files d'attente cible dans le cluster.

Procédure

Dans l'exemple, voir [Figure 54](#), à la page 410, définissez un alias de gestionnaire de files d'attente sur le gestionnaire de files d'attente de passerelle QM1 appelé DEMO:

```
DEFINE QREMOTE(DEMO) RNAME(' ') RQMNAME(' ')
```

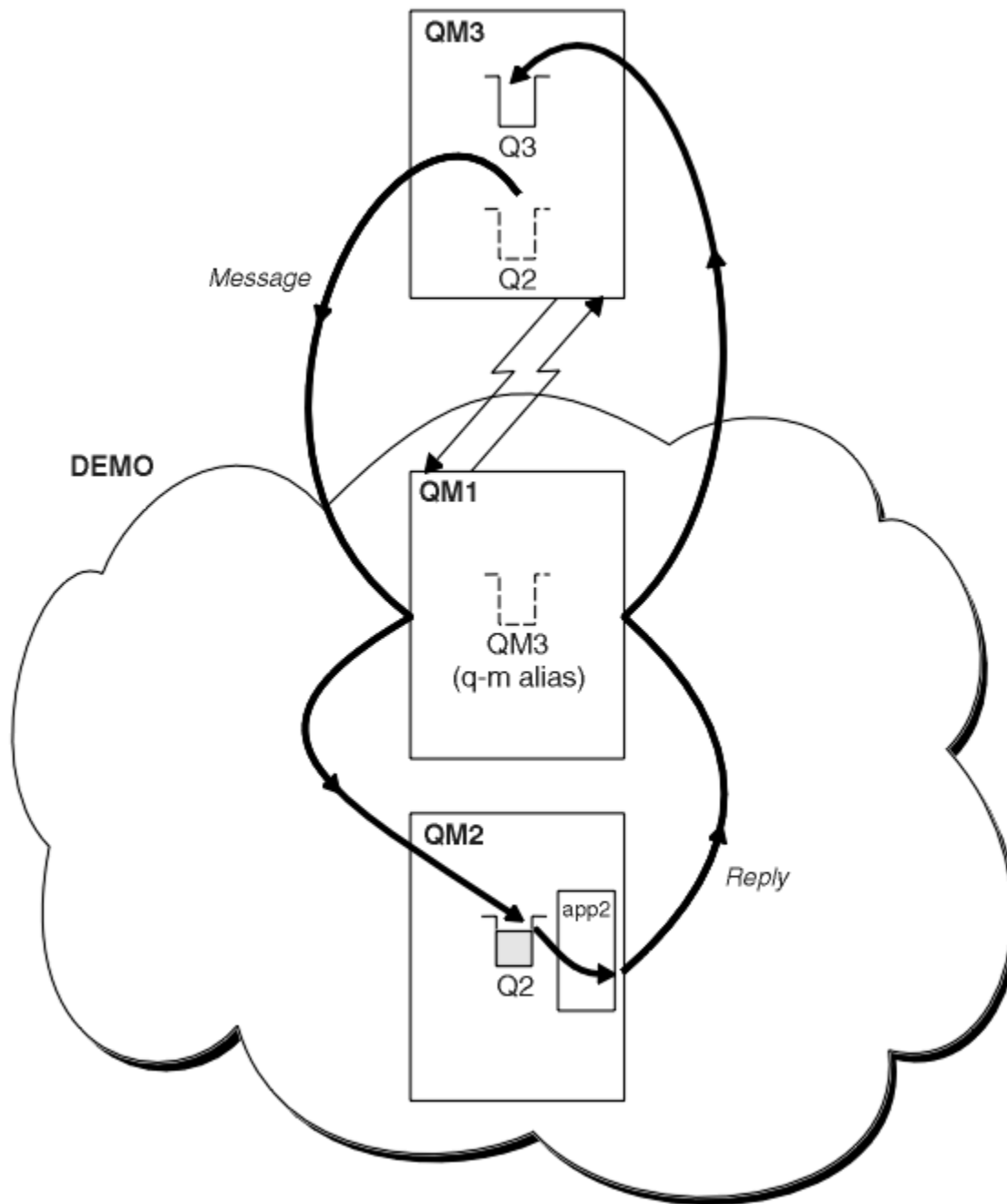


Figure 54. Insertion à partir d'un gestionnaire de files d'attente en dehors du cluster

La définition QREMOTE sur QM1 rend l'alias de gestionnaire de files d'attente DEMO connu du gestionnaire de files d'attente de passerelle. QM3, Le gestionnaire de files d'attente en dehors du cluster peut utiliser l'alias de gestionnaire de files d'attente DEMO pour envoyer des messages aux files d'attente de cluster sur DEMO, au lieu d'utiliser un nom de gestionnaire de files d'attente réel.

Si vous adoptez la convention d'utilisation du nom de cluster pour nommer la file d'attente de transmission qui se connecte à un cluster, la définition de file d'attente éloignée pour Q2 devient:

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(DEMO) XMIT(DEMO)
```

Résultats

Les messages destinés à Q2 sur DEMO sont placés dans la file d'attente de transmission DEMO . A partir de la file d'attente de transmission, ils sont transférés par le canal émetteur vers le gestionnaire de files d'attente de passerelle, QM1. Le gestionnaire de files d'attente de passerelle achemine les messages vers n'importe quel gestionnaire de files d'attente du cluster qui héberge la file d'attente de cluster Q2.

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Avant de commencer

La [Figure 55](#), à la [page 412](#) montre un gestionnaire de files d'attente, QM2, dans le cluster DEMO. Il envoie une demande à une file d'attente, Q3, hébergée sur le gestionnaire de files d'attente en dehors du cluster. Les réponses sont renvoyées à Q2 à l'adresse QM2 dans le cluster.

Pour communiquer avec le gestionnaire de files d'attente en dehors du cluster, un ou plusieurs gestionnaires de files d'attente dans le cluster agissent en tant que passerelle. Un gestionnaire de files d'attente de passerelle possède un chemin de communication vers les gestionnaires de files d'attente en dehors du cluster. Dans l'exemple, QM1 est la passerelle.

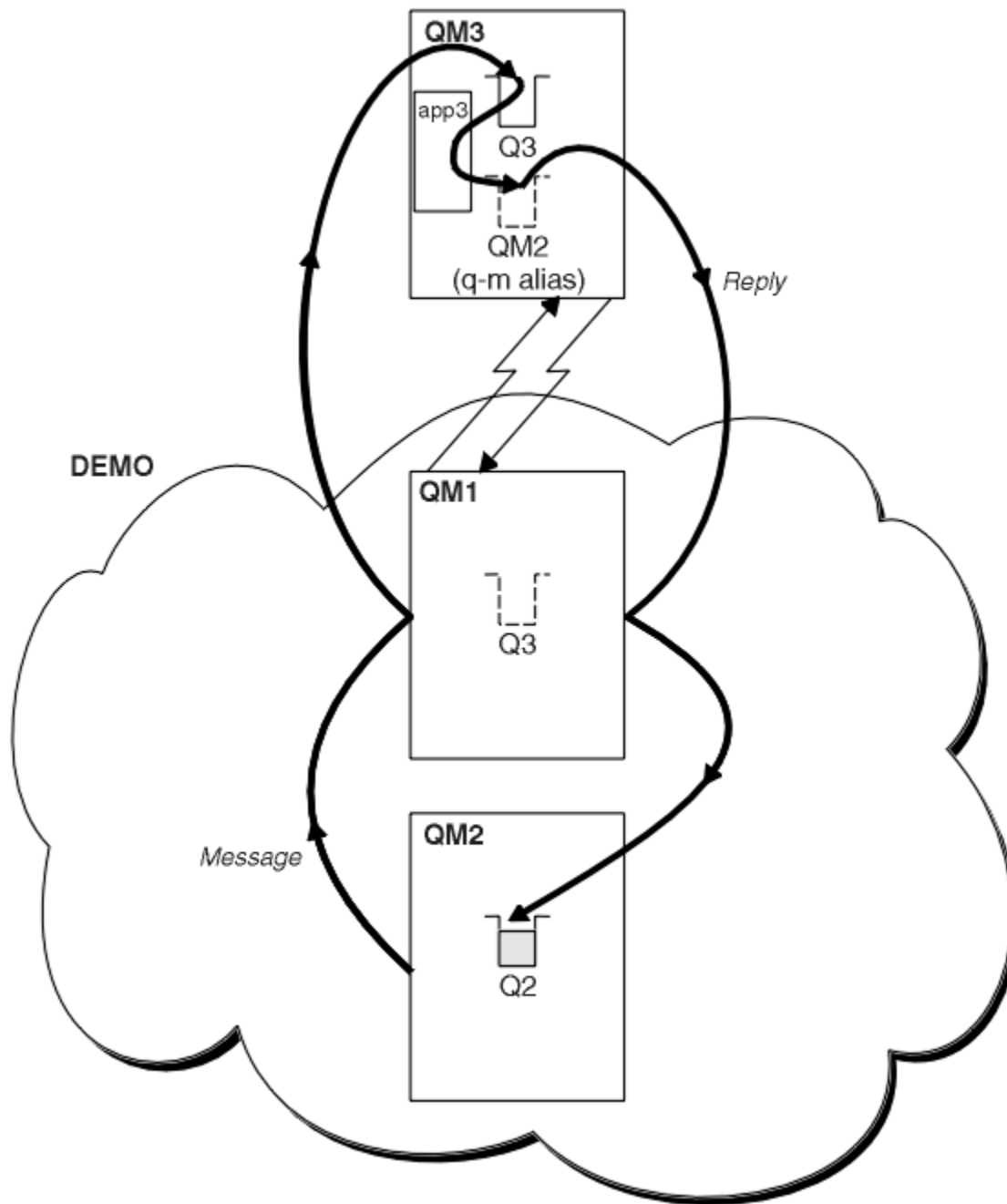


Figure 55. Insertion dans un gestionnaire de files d'attente en dehors du cluster

Pourquoi et quand exécuter cette tâche

Suivez les instructions pour configurer le chemin des messages de demande et de réponse

Procédure

1. Envoyez le message de demande à partir du cluster.

Tenez compte de la façon dont le gestionnaire de files d'attente, QM2, qui se trouve à l'intérieur du cluster, insère un message dans la file d'attente Q3 à l'adresse QM3, qui se trouve à l'extérieur du cluster.

- a) Créez une définition QREMOTE sur QM1 qui annonce la file d'attente éloignée Q3 au cluster

```
DEFINE QREMOTE(Q3) RNAME(Q3) RQMNAME(QM3) CLUSTER(DEMO)
```

Il comporte également un canal émetteur et une file d'attente de transmission vers le gestionnaire de files d'attente situé en dehors du cluster. QM3 possède un canal récepteur correspondant. Les canaux ne sont pas affichés dans la [Figure 55](#), à la page 412.

Une application sous QM2 émet un appel MQPUT spécifiant la file d'attente cible et la file d'attente à laquelle les réponses doivent être envoyées. La file d'attente cible est Q3 et la file d'attente de réponse est Q2.

Le message est envoyé à QM1, qui utilise sa définition de file d'attente éloignée pour résoudre le nom de file d'attente en Q3 dans QM3.

2. Recevez le message de réponse du gestionnaire de files d'attente en dehors du cluster.

Un gestionnaire de files d'attente en dehors du cluster doit avoir un alias de gestionnaire de files d'attente pour chaque gestionnaire de files d'attente du cluster auquel il envoie un message. L'alias du gestionnaire de files d'attente doit également indiquer le nom de la file d'attente de transmission au gestionnaire de files d'attente de passerelle. Dans cet exemple, QM3 a besoin d'une définition d'alias de gestionnaire de files d'attente pour QM2:

- a) Création d'un alias de gestionnaire de files d'attente QM2 sous QM3

```
DEFINE QREMOTE(QM2) RNAME(' ') RQMNAME(QM2) XMITQ(QM1)
```

QM3 a également besoin d'un canal émetteur et d'une file d'attente de transmission pour QM1 et QM1 a besoin d'un canal récepteur correspondant.

L'application, **app3**, sur QM3 peut ensuite envoyer des réponses à QM2, en émettant un appel MQPUT et en spécifiant le nom de la file d'attente, Q2 et le nom du gestionnaire de files d'attente, QM2.

Que faire ensuite

Vous pouvez définir plusieurs routes hors d'un cluster.

Concepts associés

[Alias de gestionnaire de files d'attente et clusters](#)

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

[Alias de file d'attente de réponse et clusters](#)

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

[Alias de file d'attente et clusters](#)

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Tâches associées

[Configuration de la demande / réponse à un cluster](#)

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

[Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster](#)

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

[Configuration des chemins d'accès aux messages entre les clusters](#)

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

Avant de commencer

Configurez l'exemple, comme illustré dans la [Figure 53, à la page 406](#) dans la «[Configuration de la demande / réponse à un cluster](#)», à la page 405.

Pourquoi et quand exécuter cette tâche

Dans ce scénario, le gestionnaire de files d'attente en dehors du cluster, QM3 dans [Figure 56, à la page 415](#), envoie des demandes à la file d'attente Q2. Q2 est hébergé sur deux gestionnaires de files d'attente, QM2 et QM4 dans le cluster DEMO. Les deux gestionnaires de files d'attente sont configurés avec l'option de liaison par défaut NOTFIXED afin d'utiliser l'équilibrage de charge. Les demandes provenant de QM3, le gestionnaire de files d'attente en dehors du cluster, sont envoyées à l'une des instances de Q2 via QM1.

QM3 ne fait pas partie d'un cluster et communique à l'aide de techniques de mise en file d'attente répartie. Il doit disposer d'un canal émetteur et d'une file d'attente de transmission vers QM1. QM1 a besoin d'un canal récepteur correspondant. Les canaux et les files d'attente de transmission ne sont pas affichés explicitement dans [Figure 56, à la page 415](#).

La procédure étend l'exemple dans [Figure 53, à la page 406](#) dans «[Configuration de la demande / réponse à un cluster](#)», à la page 405.

Procédure

1. Créez une définition QREMOTE pour Q2 sur QM3.

```
DEFINE QREMOTE(Q2) RNAME(Q2) RQMNAME(Q3) XMITQ(QM1)
```

Créez une définition QREMOTE pour chaque file d'attente du cluster dans laquelle QM3 insère des messages.

2. Créez un alias de gestionnaire de files d'attente Q3 sous QM1.

```
DEFINE QREMOTE(Q3) RNAME(' ') RQMNAME(' ')
```

Q3 n'est pas un nom de gestionnaire de files d'attente réel. Il s'agit du nom d'une définition d'alias de gestionnaire de files d'attente dans le cluster qui correspond au nom d'alias de gestionnaire de files d'attente Q3 . ' '

3. Définissez une file d'attente locale appelée Q2 sur QM2 et QM4.

```
DEFINE QLOCAL(Q2) CLUSTER(DEMO) DEFBIND(NOTFIXED)
```

4. QM1, le gestionnaire de files d'attente de passerelle, ne comporte pas de définitions spéciales.

Résultats

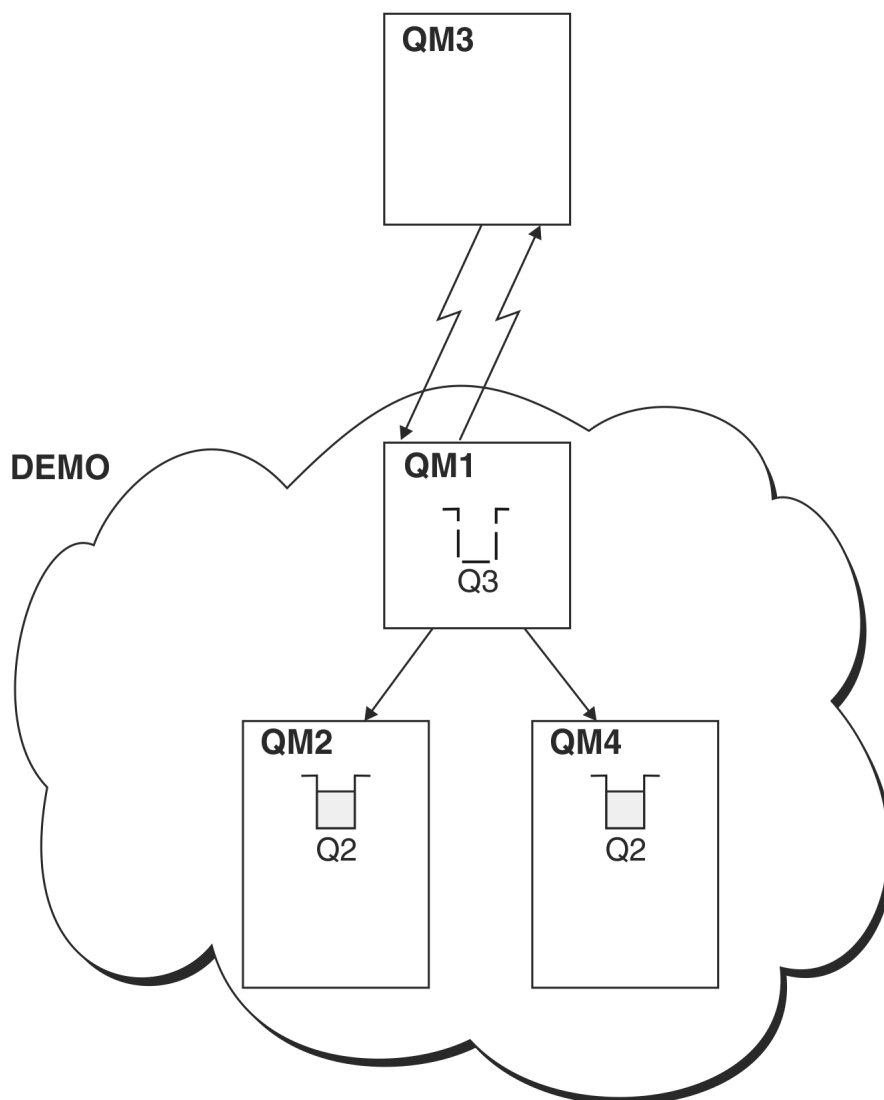


Figure 56. Insertion à partir d'un gestionnaire de files d'attente en dehors du cluster

Lorsqu'une application à l'adresse QM3 émet un appel MQPUT pour insérer un message dans Q2, la définition QREMOTE sur QM3 entraîne le routage du message via le gestionnaire de files d'attente de passerelle QM1. Lorsque QM1 reçoit le message, il est conscient que le message est toujours destiné à une file d'attente nommée Q2 et effectue la résolution de nom. QM1 vérifie ses définitions locales et n'en trouve aucune pour Q2. QM1 vérifie ensuite la configuration de son cluster et détecte qu'il a connaissance de deux instances de Q2 dans le cluster DEMO. QM1 peut désormais utiliser l'équilibrage de charge pour distribuer des messages entre les instances de Q2 résidant sur QM2 et QM4.

Concepts associés

Alias de gestionnaire de files d'attente et clusters

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

Alias de file d'attente de réponse et clusters

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Alias de file d'attente et clusters

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Résolution de nom

Tâches associées

Configuration de la demande / réponse à un cluster

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Configuration des chemins d'accès aux messages entre les clusters

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

Référence associée

Résolution de nom de file d'attente

Configuration des chemins d'accès aux messages entre les clusters

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

Pourquoi et quand exécuter cette tâche

Au lieu de regrouper tous vos gestionnaires de files d'attente dans un grand cluster, vous pouvez avoir de nombreux clusters plus petits. Chaque cluster comporte un ou plusieurs gestionnaires de files d'attente qui jouent le rôle de pont. L'avantage est que vous pouvez restreindre la visibilité des noms de file d'attente et de gestionnaire de files d'attente dans les clusters. Voir Clusters qui se chevauchent. Utilisez des alias pour modifier les noms des files d'attente et des gestionnaires de files d'attente afin d'éviter les conflits de noms ou de respecter les conventions de dénomination locales.

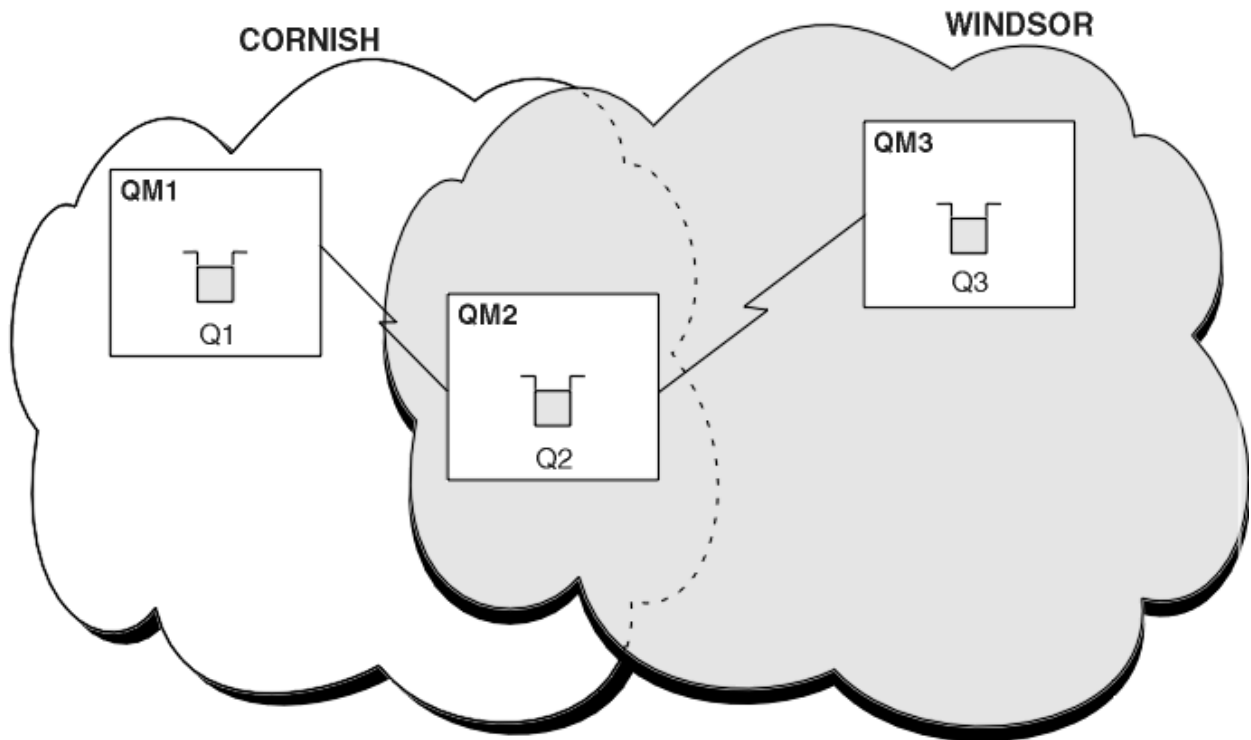


Figure 57. Pontage entre les clusters

La [Figure 57](#), à la [page 417](#) montre deux clusters avec un pont entre eux. Il pourrait y avoir plus d'un pont. Configurez les clusters à l'aide de la procédure suivante:

Procédure

1. Définissez une file d'attente de cluster, Q1 sur QM1.

```
DEFINE QLOCAL(Q1) CLUSTER(CORNISH)
```

2. Définissez une file d'attente de cluster, Q3 sur QM3.

```
DEFINE QLOCAL(Q3) CLUSTER(WINDSOR)
```

3. Créez une liste de noms appelée CORNISHWINDSOR on QM2, contenant les noms des deux clusters.

```
DEFINE NAMELIST(CORNISHWINDSOR) DESCR('CornishWindsor namelist')
NAMES(CORNISH, WINDSOR)
```

4. Définissez une file d'attente de cluster, Q2 on QM2

```
DEFINE QLOCAL(Q2) CLUSNL(CORNISHWINDSOR)
```

Que faire ensuite

QM2 est un membre des deux clusters et est la passerelle entre eux. Pour chaque file d'attente que vous souhaitez rendre visible sur le pont, vous avez besoin d'une définition QALIAS sur le pont. Par exemple, dans [Figure 57](#), à la [page 417](#), sous QM2, vous avez besoin de:

```
DEFINE QALIAS(MYQ3) TARGET(Q3) CLUSTER(CORNISH) DEFBIND(NOTFIXED)
```

A l'aide de l'alias de file d'attente, une application connectée à un gestionnaire de files d'attente dans CORNISH, par exemple QM1, peut insérer un message dans Q3. Il fait référence à Q3 en tant que MYQ3. Le message est acheminé vers Q3 à l'adresse QM3.

Lorsque vous ouvrez une file d'attente, vous devez définir DEFBIND sur NOTFIXED ou QDEF. Si DEFBIND reste la valeur par défaut, OPEN, le gestionnaire de files d'attente résout la définition d'alias en gestionnaire de files d'attente de pont qui l'héberge. Le pont ne transmet pas le message.

Pour chaque gestionnaire de files d'attente que vous souhaitez rendre visible, vous avez besoin d'une définition d'alias de gestionnaire de files d'attente. Par exemple, sous QM2, vous avez besoin de:

```
DEFINE QREMOTE(QM1) RNAME(' ') RQMNAME(QM1) CLUSTER(WINDSOR)
```

Une application connectée à un gestionnaire de files d'attente dans WINDSOR, par exemple QM3, peut insérer un message dans n'importe quelle file d'attente dans QM1, en nommant QM1 explicitement dans l'appel MQOPEN.

Concepts associés

Alias de gestionnaire de files d'attente et clusters

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

Alias de file d'attente de réponse et clusters

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Alias de file d'attente et clusters

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Tâches associées

Configuration de la demande / réponse à un cluster

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

Alias de gestionnaire de files d'attente et clusters

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

Les alias de gestionnaire de files d'attente, qui sont créés à l'aide d'une définition de file d'attente éloignée avec un RNAMEvide, ont cinq utilisations:

Remappage du nom du gestionnaire de files d'attente lors de l'envoi de messages

Un alias de gestionnaire de files d'attente peut être utilisé pour remapper le nom de gestionnaire de files d'attente spécifié dans un appel MQOPEN à un autre gestionnaire de files d'attente. Il peut s'agir

d'un gestionnaire de files d'attente de cluster. Par exemple, un gestionnaire de files d'attente peut avoir la définition d'alias de gestionnaire de files d'attente:

```
DEFINE QREMOTE(YORK) RNAME(' ') RQMNAME(CLUSQM)
```

YORK peut être utilisé comme alias du gestionnaire de files d'attente appelé CLUSQM. Lorsqu'une application du gestionnaire de files d'attente qui a créé cette définition insère un message dans le gestionnaire de files d'attente YORK, le gestionnaire de files d'attente local résout le nom en CLUSQM. Si le gestionnaire de files d'attente local n'est pas appelé CLUSQM, il place le message dans la file d'attente de transmission du cluster à déplacer vers CLUSQM. Elle modifie également l'en-tête de transmission pour indiquer CLUSQM au lieu de YORK.

Remarque : La définition s'applique uniquement au gestionnaire de files d'attente qui l'a créée. Pour annoncer l'alias à l'ensemble du cluster, vous devez ajouter l'attribut CLUSTER à la définition de file d'attente éloignée. Les messages des autres gestionnaires de files d'attente destinés à YORK sont ensuite envoyés à CLUSQM.

Modification ou spécification de la file d'attente de transmission lors de l'envoi de messages

L'utilisation d'alias peut être utilisée pour joindre un cluster à un système non-cluster. Par exemple, les gestionnaires de files d'attente du cluster ITALY peuvent communiquer avec le gestionnaire de files d'attente appelé PALERMO, qui se trouve en dehors du cluster. Pour communiquer, l'un des gestionnaires de files d'attente du cluster doit agir en tant que passerelle. A partir du gestionnaire de files d'attente de passerelle, exécutez la commande suivante:

```
DEFINE QREMOTE(ROME) RNAME(' ') RQMNAME(PALERMO) XMITQ(X) CLUSTER(ITALY)
```

La commande est une définition d'alias de gestionnaire de files d'attente. Il définit et annonce ROME en tant que gestionnaire de files d'attente sur lequel les messages provenant de n'importe quel gestionnaire de files d'attente du cluster ITALY peuvent effectuer plusieurs étapes pour atteindre leur destination dans PALERMO. Les messages insérés dans une file d'attente ouverte avec le nom de gestionnaire de files d'attente défini sur ROME sont envoyés au gestionnaire de files d'attente de passerelle avec la définition d'alias de gestionnaire de files d'attente. Une fois sur place, les messages sont placés dans la file d'attente de transmission X et déplacés par des canaux non-cluster vers le gestionnaire de files d'attente PALERMO.

Le choix du nom ROME dans cet exemple n'est pas significatif. Les valeurs de QREMOTE et de RQMNAME peuvent être identiques.

Détermination de la destination lors de la réception de messages

Lorsqu'un gestionnaire de files d'attente reçoit un message, il extrait le nom de la file d'attente de destination et du gestionnaire de files d'attente de l'en-tête de transmission. Il recherche une définition d'alias de gestionnaire de files d'attente portant le même nom que le gestionnaire de files d'attente dans l'en-tête de transmission. S'il en trouve un, il remplace la valeur RQMNAME de la définition d'alias de gestionnaire de files d'attente par le nom du gestionnaire de files d'attente dans l'en-tête de transmission.

Il existe deux raisons d'utiliser un alias de gestionnaire de files d'attente de cette manière:

- Pour diriger des messages vers un autre gestionnaire de files d'attente
- Pour modifier le nom du gestionnaire de files d'attente afin qu'il soit identique à celui du gestionnaire de files d'attente local

Utilisation d'alias de gestionnaire de files d'attente dans un gestionnaire de files d'attente de passerelle pour acheminer les messages entre les gestionnaires de files d'attente de différents clusters.

Une application peut envoyer un message à une file d'attente dans un autre cluster à l'aide d'un alias de gestionnaire de files d'attente. La file d'attente n'a pas besoin d'être une file d'attente de cluster. La file d'attente est définie dans un cluster. L'application est connectée à un gestionnaire de files d'attente dans un autre cluster. Un gestionnaire de files d'attente de passerelle connecte les deux

clusters. Si la file d'attente n'est pas définie comme étant en cluster, pour que le routage correct soit effectué, l'application doit ouvrir la file d'attente à l'aide du nom de la file d'attente et d'un nom d'alias de gestionnaire de files d'attente en cluster. Pour un exemple de configuration, voir «Création de deux clusters qui se chevauchent avec un gestionnaire de files d'attente de passerelle», à la page 367, à partir duquel le flux de messages de réponse illustré dans la figure 1 est pris.

Le diagramme montre le chemin emprunté par le message de réponse vers une file d'attente dynamique temporaire, appelée RQ. L'application serveur, connectée à QM3, ouvre la file d'attente de réponses à l'aide du nom de gestionnaire de files d'attente QM2. Le nom du gestionnaire de files d'attente QM2 est défini en tant qu'alias de gestionnaire de files d'attente en cluster sous QM1. QM3 achemine le message de réponse vers QM1. QM1 achemine le message vers QM2.

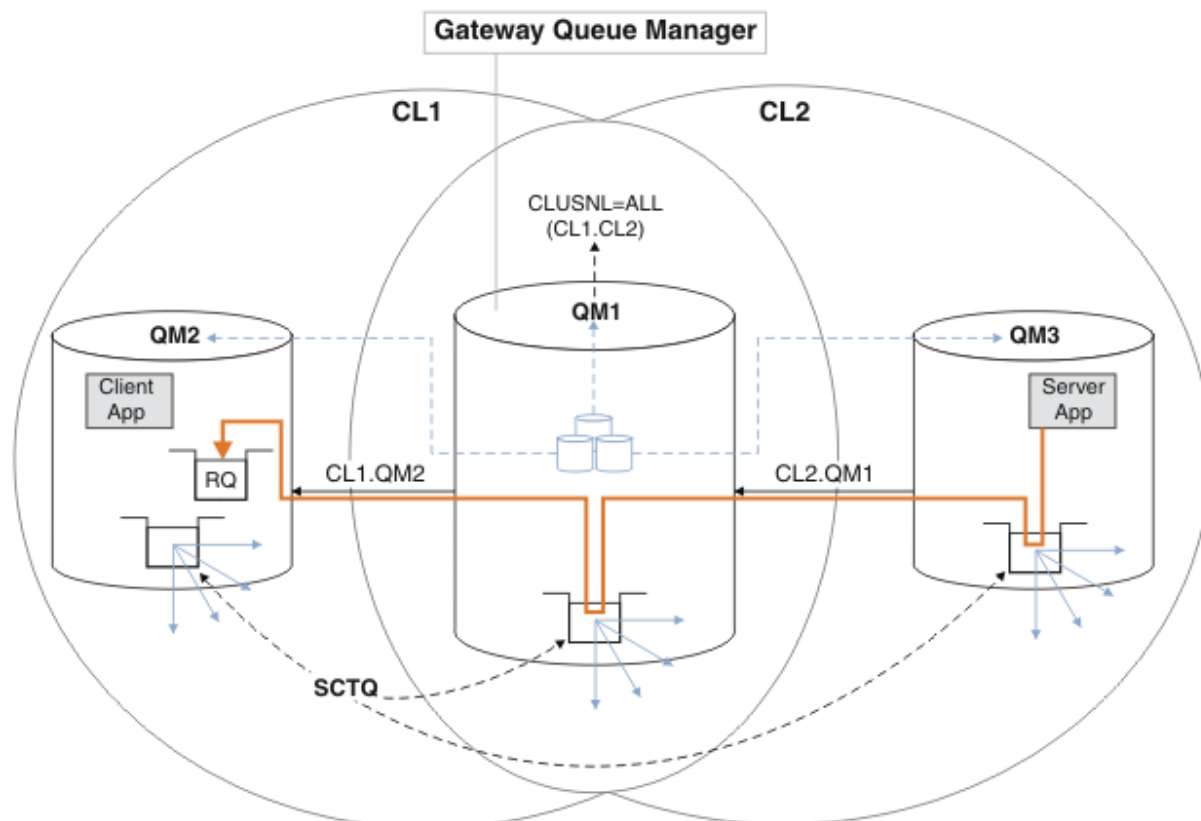


Figure 58. Utilisation d'un alias de gestionnaire de files d'attente pour renvoyer le message de réponse à un autre cluster

Le mode de fonctionnement du routage est le suivant. Chaque gestionnaire de files d'attente de chaque cluster possède une définition d'alias de gestionnaire de files d'attente sous QM1. Les alias sont regroupés dans tous les clusters. Les flèches en pointillés grises de chacun des alias vers un gestionnaire de files d'attente indiquent que chaque alias de gestionnaire de files d'attente est résolu en gestionnaire de files d'attente réel dans au moins un des clusters. Dans ce cas, l'alias QM2 est mis en cluster à la fois dans le cluster CL1 et dans CL2, et est résolu en gestionnaire de files d'attente réel QM2 dans CL1. L'application serveur crée le message de réponse en utilisant le nom de la file d'attente de réponse RQ et le nom du gestionnaire de files d'attente de réponse QM2. Le message est acheminé vers QM1 car la définition d'alias de gestionnaire de files d'attente QM2 est définie sur QM1 dans le cluster CL2 et le gestionnaire de files d'attente QM2 ne se trouve pas dans le cluster CL2. Comme le message ne peut pas être envoyé au gestionnaire de files d'attente cible, il est envoyé au gestionnaire de files d'attente qui possède la définition d'alias.

QM1 place le message dans la file d'attente de transmission du cluster sur QM1 pour le transfert vers QM2. QM1 route le message vers QM2 car la définition d'alias de gestionnaire de files d'attente sous QM1 for QM2 définit QM2 comme le gestionnaire de files d'attente cible réel. La définition n'est pas circulaire, car les définitions d'alias ne peuvent faire référence qu'à des définitions réelles ; l'alias

ne peut pas pointer vers lui-même. La définition réelle est résolue par QM1, car QM1 et QM2 se trouvent dans le même cluster, CL1. QM1 recherche les informations de connexion pour QM2 dans le référentiel pour CL1 et achemine le message vers QM2. Pour que le message soit réacheminé par QM1, l'application serveur doit avoir ouvert la file d'attente de réponses avec l'option DEFBIND définie sur MQBND_BIND_NOT_FIXED. Si l'application serveur a ouvert la file d'attente de réponses avec l'option MQBND_BIND_ON_OPEN, le message n'est pas réacheminé et se retrouve dans une file d'attente de messages non livrés.

Utilisation d'un gestionnaire de files d'attente en tant que passerelle dans le cluster pour équilibrer la charge de travail des messages provenant de l'extérieur du cluster.

Vous définissez une file d'attente appelée EDINBURGH sur plusieurs gestionnaires de files d'attente du cluster. Vous souhaitez que le mécanisme de mise en cluster équilibre la charge de travail pour les messages provenant de l'extérieur du cluster qui arrivent dans cette file d'attente.

Un gestionnaire de files d'attente externe au cluster a besoin d'une file d'attente de transmission et d'un canal émetteur vers un gestionnaire de files d'attente du cluster. Cette file d'attente est appelée gestionnaire de files d'attente de passerelle. Pour tirer parti du mécanisme d'équilibrage de charge par défaut, l'une des règles suivantes doit s'appliquer:

- Le gestionnaire de files d'attente de passerelle ne doit pas contenir d'instance de la file d'attente EDINBURGH.
- Le gestionnaire de files d'attente de passerelle spécifie CLWLUSEQ (ANY) sous ALTER QMGR.

Pour un exemple d'équilibrage de charge externe à un cluster, voir [«Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster»](#), à la page 414

Concepts associés

Alias de file d'attente de réponse et clusters

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Alias de file d'attente et clusters

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Tâches associées

Configuration de la demande / réponse à un cluster

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

Configuration des chemins d'accès aux messages entre les clusters

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

Alias de file d'attente de réponse et clusters

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Exemple :

- Une application au niveau du gestionnaire de files d'attente VENICE envoie un message au gestionnaire de files d'attente PISA à l'aide de l'appel MQPUT . L'application fournit les informations de file d'attente de réponse suivantes dans le descripteur de message:

```
ReplyToQ=' QUEUE '  
ReplyToQMgtr=' '
```

- Pour que les réponses envoyées à QUEUE puissent être reçues sur OTHERQ à l'adresse PISA, créez une définition de file d'attente éloignée sur VENICE qui soit utilisée comme alias de file d'attente de réponse. L'alias n'est effectif que sur le système sur lequel il a été créé.

```
DEFINE QREMOTE(Queue) RNAME(OTHERQ) RQMNAME(PISA)
```

RQMNAME et QREMOTE peuvent spécifier les mêmes noms, même si RQMNAME est lui-même un gestionnaire de files d'attente de cluster.

Concepts associés

Alias de gestionnaire de files d'attente et clusters

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

Alias de file d'attente et clusters

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Tâches associées

Configuration de la demande / réponse à un cluster

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

Configuration des chemins d'accès aux messages entre les clusters

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

Alias de file d'attente et clusters

Utilisez des alias de file d'attente pour masquer le nom d'une file d'attente de cluster, pour mettre en cluster une file d'attente, adopter des attributs différents ou adopter des contrôles d'accès différents.

Une définition QALIAS est utilisée pour créer un alias permettant de connaître une file d'attente. Vous pouvez créer un alias pour plusieurs raisons:

- Vous souhaitez commencer à utiliser une autre file d'attente, mais vous ne souhaitez pas modifier vos applications.
- Vous ne voulez pas que les applications connaissent le nom réel de la file d'attente dans laquelle elles placent des messages.
- Vous pouvez avoir une convention de dénomination différente de celle dans laquelle la file d'attente est définie.
- Il se peut que vos applications ne soient pas autorisées à accéder à la file d'attente par son nom réel, mais uniquement par son alias.

Créez une définition QALIAS sur un gestionnaire de files d'attente à l'aide de la commande DEFINE QALIAS . Par exemple, exécutez la commande suivante:

```
DEFINE QALIAS(PUBLIC) TARGET(LOCAL) CLUSTER(C)
```

La commande annonce une file d'attente appelée PUBLIC aux gestionnaires de files d'attente du cluster C. PUBLIC est un alias qui se résout dans la file d'attente appelée LOCAL. Les messages envoyés à PUBLIC sont acheminés vers la file d'attente appelée LOCAL.

Vous pouvez également utiliser une définition d'alias de file d'attente pour résoudre un nom de file d'attente en file d'attente de cluster. Par exemple, exécutez la commande suivante:

```
DEFINE QALIAS(PRIVATE) TARGET(PUBLIC)
```

La commande permet à un gestionnaire de files d'attente d'utiliser le nom PRIVATE pour accéder à une file d'attente annoncée ailleurs dans le cluster sous le nom PUBLIC. Etant donné que cette définition n'inclut pas l'attribut CLUSTER , elle s'applique uniquement au gestionnaire de files d'attente qui l'utilise.

Concepts associés

Alias de gestionnaire de files d'attente et clusters

Utilisez des alias de gestionnaire de files d'attente pour masquer le nom des gestionnaires de files d'attente lors de l'envoi de messages à destination ou en dehors d'un cluster, et pour équilibrer la charge de travail des messages envoyés à un cluster.

Alias de file d'attente de réponse et clusters

Une définition d'alias de file d'attente de réponse permet de spécifier des noms alternatifs pour les informations de réponse. Les définitions d'alias de file d'attente de réponse peuvent être utilisées avec des clusters identiques à ceux d'un environnement de mise en file d'attente répartie.

Tâches associées

Configuration de la demande / réponse à un cluster

Configurez un chemin de message de demande / réponse à partir d'un gestionnaire de files d'attente en dehors d'un cluster. Masquez les détails internes du cluster en utilisant un gestionnaire de files d'attente de passerelle comme chemin de communication vers et depuis le cluster.

Configuration d'une demande / réponse à partir d'un cluster

Configurez un chemin de message de demande / réponse à partir d'un cluster vers un gestionnaire de files d'attente en dehors du cluster. Masquez les détails de la façon dont un gestionnaire de files d'attente à l'intérieur du cluster communique en dehors du cluster à l'aide d'un gestionnaire de files d'attente de passerelle.

Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.


Configuration des chemins d'accès aux messages entre les clusters

Connectez des clusters à l'aide d'un gestionnaire de files d'attente de passerelle. Rendre les files d'attente ou les gestionnaires de files d'attente visibles pour tous les clusters en définissant des alias de file d'attente de cluster ou de gestionnaire de files d'attente de cluster sur le gestionnaire de files d'attente de passerelle.

Utilisation de clusters pour la gestion de charge de travail

En définissant plusieurs instances d'une file d'attente sur différents gestionnaires de files d'attente d'un cluster, vous pouvez répartir le travail de maintenance de la file d'attente sur plusieurs serveurs. Plusieurs facteurs peuvent empêcher la remise en file d'attente des messages dans un gestionnaire de files d'attente différent en cas d'échec.

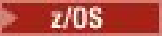
En plus de configurer des clusters pour réduire l'administration du système, vous pouvez créer des clusters dans lesquels plusieurs gestionnaires de files d'attente hébergent une instance de la même file d'attente.

Vous pouvez organiser votre cluster de sorte que les gestionnaires de files d'attente qu'il contient soient des clones les uns des autres. Chaque gestionnaire de files d'attente peut exécuter les mêmes applications et disposer de définitions locales des mêmes files d'attente.  Par exemple, dans un sysplex parallèle z/OS, les applications clonées peuvent accéder aux données d'une base de données Db2 partagée ou VSAM (Virtual Storage Access Method). Vous pouvez répartir la charge de travail entre vos gestionnaires de files d'attente en disposant de plusieurs instances d'une application. Chaque instance de l'application reçoit des messages et s'exécute indépendamment des autres.

Les avantages de l'utilisation de clusters de cette manière sont les suivants:

- Disponibilité accrue de vos files d'attente et de vos applications.
- Débit plus rapide des messages.
- Distribution plus uniforme de la charge de travail dans votre réseau.

N'importe lequel des gestionnaires de files d'attente qui héberge une instance d'une file d'attente particulière peut traiter les messages destinés à cette file d'attente, et les applications ne nomment pas de gestionnaire de files d'attente lors de l'envoi de messages. Si un cluster contient plusieurs instances de la même file d'attente, IBM MQ sélectionne un gestionnaire de files d'attente vers lequel acheminer un message. Les destinations appropriées sont choisies en fonction de la disponibilité du gestionnaire de files d'attente et de la file d'attente et d'un certain nombre d'attributs spécifiques à la charge de travail du cluster associés aux gestionnaires de files d'attente, aux files d'attente et aux canaux. Voir [Équilibrage de la charge de travail dans les clusters](#).

 Dans IBM MQ for z/OS, les gestionnaires de files d'attente qui se trouvent dans des groupes de partage de files d'attente peuvent héberger des files d'attente de cluster en tant que files d'attente partagées. Les files d'attente de cluster partagées sont disponibles pour tous les gestionnaires de files d'attente du même groupe de partage de files d'attente. Par exemple, dans [Un cluster avec plusieurs instances de la même file d'attente](#), l'un ou les deux gestionnaires de files d'attente QM2 et QM4 peuvent être un gestionnaire de files d'attente partagées. Chacune possède une définition pour la file d'attente Q3. Tous les gestionnaires de files d'attente du même groupe de partage de files d'attente que QM4 peuvent lire un message inséré dans la file d'attente partagée Q3. Chaque groupe de partage de files d'attente peut contenir jusqu'à 32 gestionnaires de files d'attente, chacun ayant accès aux mêmes données. Le partage de files d'attente augmente considérablement le débit de vos messages.

Pour plus d'informations sur les configurations de cluster pour la gestion de la charge de travail, voir les sous-rubriques suivantes:

Concepts associés

[Comparaison de la mise en cluster et de la mise en file d'attente répartie](#)

[Mise en file d'attente répartie et clusters](#)

[Composants d'un cluster](#)

[Canaux de cluster](#)

[Que se passe-t-il si une file d'attente de cluster est désactivée pour MQPUT?](#)

[L'équilibrage de charge défini sur un canal émetteur de cluster ne fonctionne pas](#)

[«Routage des messages vers et depuis des clusters», à la page 404](#)

Utilisez des alias de file d'attente, des alias de gestionnaire de files d'attente et des définitions de file d'attente éloignée pour connecter des clusters à des gestionnaires de files d'attente externes et à d'autres clusters.

Tâches associées

Écriture et compilation des exits de charge de travail de cluster

«Configuration d'un cluster de gestionnaires de files d'attente», à la page 316

Les clusters fournissent un mécanisme d'interconnexion des gestionnaires de files d'attente qui simplifie à la fois la configuration initiale et la gestion en cours. Vous pouvez définir des composants de cluster et créer et gérer des clusters.

«Configuration d'un nouveau cluster», à la page 331

Suivez ces instructions pour configurer l'exemple de cluster. Des instructions distinctes décrivent la configuration du cluster sur TCP/IP, LU 6.2 et avec une ou plusieurs files d'attente de transmission. Testez le cluster en envoyant un message d'un gestionnaire de files d'attente à l'autre.

«Configuration de l'équilibrage de charge depuis l'extérieur d'un cluster», à la page 414

Configurez un chemin de message à partir d'un gestionnaire de files d'attente en dehors d'un cluster vers n'importe quelle copie d'une file d'attente de cluster. Le résultat est l'équilibrage de la charge de travail des demandes provenant de l'extérieur du cluster vers chaque instance d'une file d'attente de cluster.

Référence associée

Exemple de programme de surveillance de file d'attente de cluster (AMQSCLM)

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Figure 59, à la page 426 montre un cluster dans lequel il existe plusieurs définitions pour la file d'attente Q3. Si une application à l'adresse QM1 insère un message dans Q3, elle ne sait pas nécessairement quelle instance de Q3 va traiter son message. Si une application s'exécute sous QM2 ou QM4, où il existe des instances locales de Q3, l'instance locale de Q3 est ouverte par défaut. En définissant l'attribut de file d'attente CLWLUSEQ, l'instance locale de la file d'attente peut être traitée comme une instance éloignée de la file d'attente.

L'option MQOPEN DefBind contrôle si le gestionnaire de files d'attente cible est choisi lorsque l'appel MQOPEN est émis ou lorsque le message est transféré à partir de la file d'attente de transmission.

Si vous définissez DefBind sur MQBND_BIND_NOT_FIXED, le message peut être envoyé à une instance de la file d'attente qui est disponible lorsque le message est transmis. Cela évite les problèmes suivants:

- La file d'attente cible n'est pas disponible lorsque le message arrive sur le gestionnaire de files d'attente cible.
- L'état de la file d'attente a changé.
- Le message a été inséré à l'aide d'un alias de file d'attente de cluster et aucune instance de la file d'attente cible n'existe sur le gestionnaire de files d'attente où l'instance de l'alias de file d'attente de cluster est définie.

Si ces problèmes sont détectés lors de la transmission, une autre instance disponible de la file d'attente cible est recherchée et le message est réacheminé. Si aucune instance de la file d'attente n'est disponible, le message est placé dans la file d'attente de rebut.

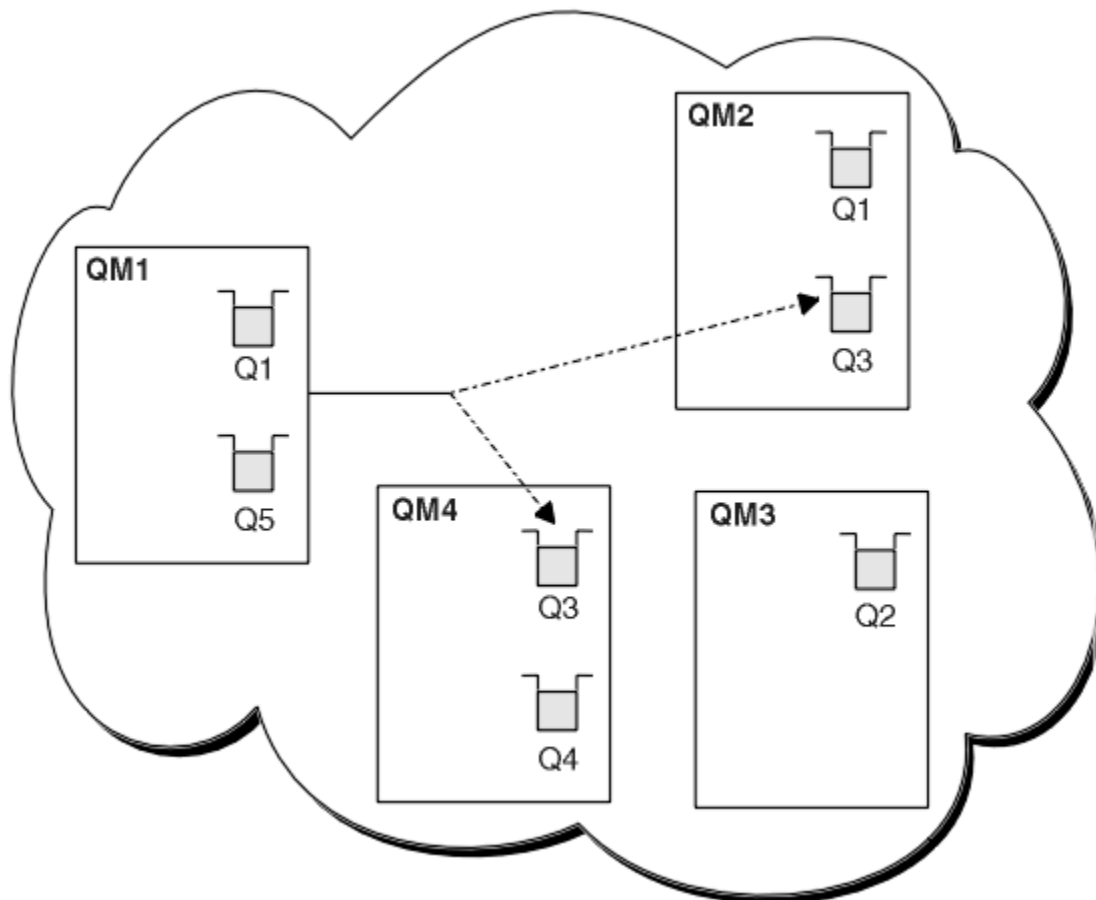


Figure 59. Un cluster avec plusieurs instances de la même file d'attente

L'un des facteurs qui peut empêcher le réacheminement des messages est l'affectation de messages à un gestionnaire de files d'attente fixe ou à un canal avec `MQBND_BIND_ON_OPEN`. Les messages liés à `MQOPEN` ne sont jamais réalloués à un autre canal. Notez également que la réallocation de messages n'a lieu que lorsqu'un canal de cluster est en fait défaillant. La réallocation n'a pas lieu si le canal a déjà échoué.

Le système tente de rediriger un message si le gestionnaire de files d'attente de destination est mis hors service. Ce faisant, il n'affecte pas l'intégrité du message en risquant de le perdre ou en créant un doublon. Si un gestionnaire de files d'attente échoue et laisse un message en attente de validation, ce message n'est pas réacheminé.

z/OS Sous IBM MQ for z/OS, le canal ne s'arrête pas complètement tant que le processus de réallocation des messages n'est pas terminé. L'arrêt du canal avec le mode défini sur `FORCE` ou `TERMINATE` interrompt le processus. Par conséquent, si vous effectuez cette opération, certains messages `BIND_NOT_FIXED` ont peut-être déjà été réalloués à un autre canal ou les messages sont peut-être dans le désordre.

Remarque : **z/OS**

1. Avant de configurer un cluster comportant plusieurs instances de la même file d'attente, assurez-vous que vos messages n'ont pas de dépendances les uns par rapport aux autres. Par exemple, vous avez besoin d'être traité dans une séquence spécifique ou par le même gestionnaire de files d'attente.
2. Faites en sorte que les définitions des différentes instances de la même file d'attente soient identiques. Sinon, vous obtenez des résultats différents à partir de différents appels `MQINQ`.

Concepts associés

[Programmation d'application et clusters](#)

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans Ajout d'un nouveau gestionnaire de files d'attente à un cluster. Il contient trois gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets, PARIS un référentiel partiel. L'application d'inventaire s'exécute sur le système à New York, connecté au gestionnaire de files d'attente NEWYORK . L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ .
- Nous souhaitons ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour ajouter un gestionnaire de files d'attente qui héberge une file d'attente en local.

Procédure

1. Modifiez le gestionnaire de files d'attente PARIS .

Pour que l'application à Paris puisse utiliser INVENTQ à Paris et celle à New York, nous devons en informer le gestionnaire de files d'attente. Sous PARIS , exécutez la commande suivante:

```
ALTER QMGR CLWLUSEQ(ANY)
```

2. Recherchez les affinités de message dans l'application d'inventaire.

Avant de continuer, assurez-vous que l'application d'inventaire ne possède pas de dépendances sur la séquence de traitement des messages. Pour plus d'informations, voir [Gestion des affinités de message](#).

3. Installez l'application d'inventaire sur le système à Paris.
4. Définissez la file d'attente de cluster INVENTQ.

La file d'attente INVENTQ déjà hébergée par le gestionnaire de files d'attente NEWYORK doit également être hébergée par PARIS. Définissez-le sur le gestionnaire de files d'attente PARIS comme suit:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

z/OS Maintenant que vous avez terminé toutes les définitions, si ce n'est pas déjà fait, démarrez l'initiateur de canal sur IBM MQ for z/OS.

Sur toutes les plateformes, démarrez un programme d'écoute sur le gestionnaire de files d'attente PARIS. Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Résultats

La Figure 60, à la page 428 présente le cluster configuré par cette tâche.

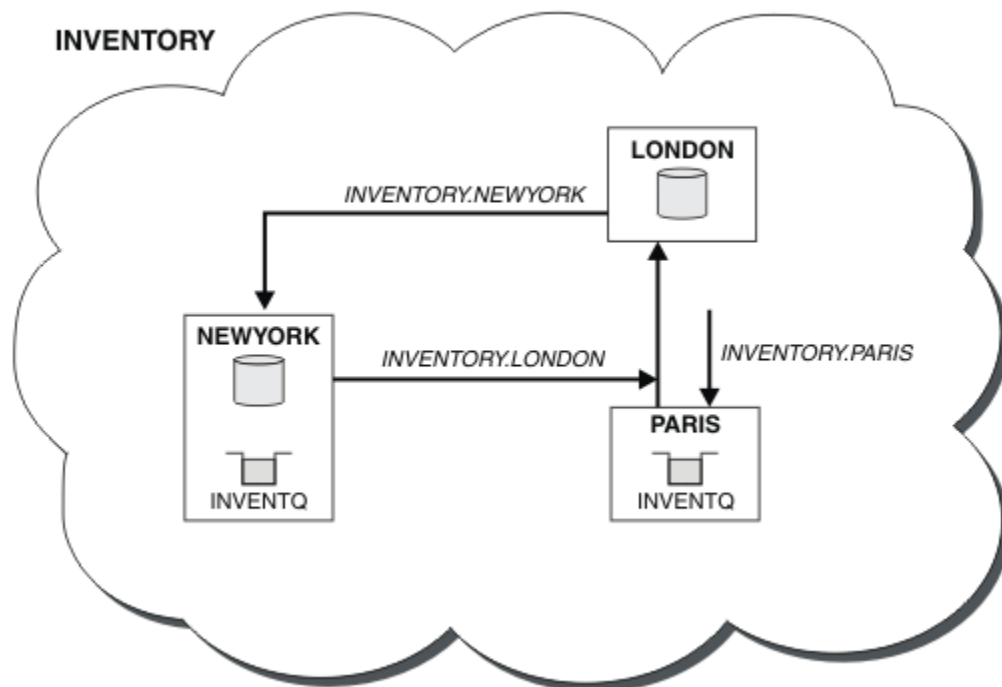


Figure 60. Le cluster INVENTORY, avec trois gestionnaires de files d'attente

La modification de ce cluster a été effectuée sans que vous ayez à modifier les gestionnaires de files d'attente NEWYORK ou LONDON. Les référentiels complets de ces gestionnaires de files d'attente sont mis à jour automatiquement avec les informations dont ils ont besoin pour pouvoir envoyer des messages à INVENTQ à l'adresse PARIS.

Que faire ensuite

La file d'attente INVENTQ et l'application d'inventaire sont désormais hébergées sur deux gestionnaires de files d'attente du cluster. Cela augmente leur disponibilité, accélère le débit des messages et permet

de répartir la charge de travail entre les deux gestionnaires de files d'attente. Les messages insérés dans INVENTQ par l'un des gestionnaires de files d'attente LONDON, NEWYORK, PARIS sont routés alternativement vers PARIS ou NEWYORK, de sorte que la charge de travail soit équilibrée.

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans "Ajout d'un gestionnaire de files d'attente à un cluster". Il contient trois gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets, PARIS un référentiel partiel. L'application d'inventaire s'exécute sur le système à New York, connecté au gestionnaire de files d'attente NEWYORK . L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ .
- Un nouveau magasin est en cours d'ajout dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour utiliser deux réseaux dans un cluster.

Procédure

1. Choisissez le référentiel complet auquel TOKYO fait référence en premier.

Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets pour collecter des informations sur le cluster. Il crée son propre référentiel partiel. Il n'a pas de signification particulière pour le référentiel que vous choisissez. Dans cet exemple, NEWYORK est choisi. Une fois que le nouveau gestionnaire de files d'attente a rejoint le cluster, il communique avec les deux référentiels.

2. Définissez les canaux CLUSRCVR .

Chaque gestionnaire de files d'attente d'un cluster doit définir un récepteur de cluster sur lequel il peut recevoir des messages. Ce gestionnaire de files d'attente doit pouvoir communiquer sur chaque réseau.


```
DEFINE CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME('TOKYO.NETB.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel using network B for TOKYO')
```

```
DEFINE CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME('TOKYO.NETA.CMSTORE.COM') CLUSTER(INVENTORY) DESCR('Cluster-receiver
channel using network A for TOKYO')
```

3. Définissez un canal CLUSSDR sur le gestionnaire de files d'attente TOKYO .

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal émetteur de cluster sur lequel il peut envoyer des messages à son premier référentiel complet. Dans ce cas, nous avons choisi NEWYORK, de sorte que TOKYO a besoin de la définition suivante:

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY) DESCR('Cluster-sender
channel from TOKYO to repository at NEWYORK')
```

 Maintenant que vous avez terminé toutes les définitions, si ce n'est pas déjà fait, démarrez l'initiateur de canal sur IBM MQ for z/OS.

Sur toutes les plateformes, démarrez un programme d'écoute sur le gestionnaire de files d'attente PARIS. Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Résultats

La [Figure 61](#), à la page 431 présente le cluster configuré par cette tâche.

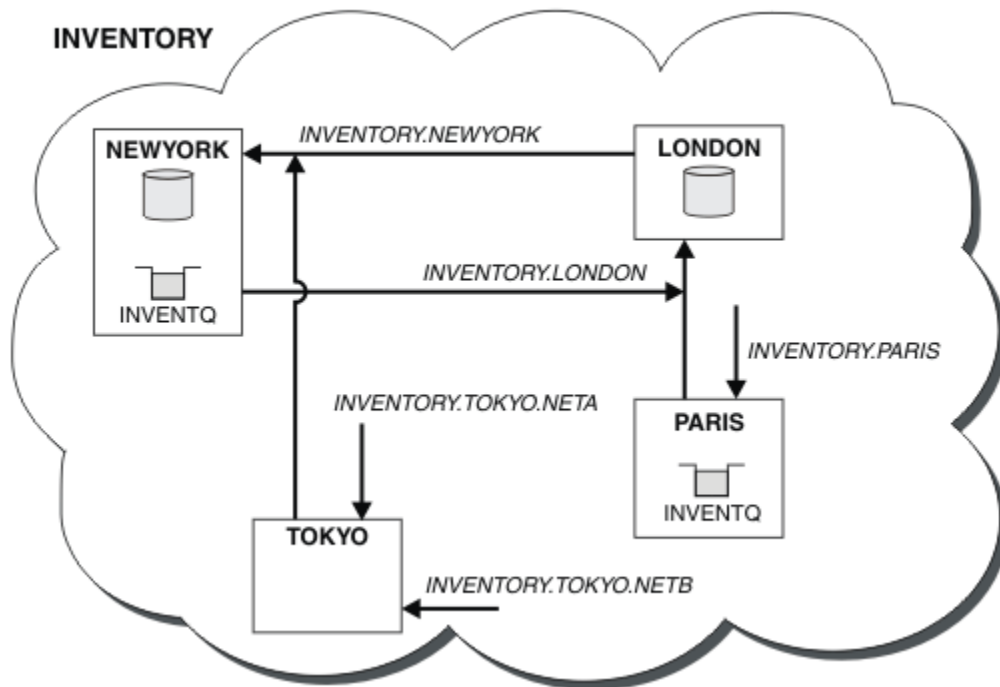


Figure 61. Le cluster INVENTORY , avec quatre gestionnaires de files d'attente

En n'effectuant que trois définitions, nous avons ajouté le gestionnaire de files d'attente TOKYO au cluster avec deux routes réseau différentes disponibles.

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

«Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343

Suivez ces instructions pour ajouter un gestionnaire de files d'attente au cluster que vous avez créé. Les messages vers les files d'attente de cluster et les rubriques sont transférés à l'aide de la file d'attente de transmission de cluster unique SYSTEM . CLUSTER . TRANSMIT . QUEUE.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans «Utilisation de deux réseaux dans un cluster», à la page 429. Il contient quatre gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets ; PARIS et TOKYO contiennent des référentiels partiels. L'application d'inventaire s'exécute sur le système de New York, connecté au gestionnaire de files d'attente NEWYORK. Le gestionnaire de files d'attente TOKYO possède deux réseaux différents sur lesquels il peut communiquer.
- Vous souhaitez faire de l'un des réseaux le réseau principal et de l'autre le réseau de secours. Vous prévoyez d'utiliser le réseau de secours en cas de problème avec le réseau principal.

Pourquoi et quand exécuter cette tâche

Utilisez l'attribut NETPRTY pour configurer un réseau principal et un réseau secondaire dans un cluster.

Procédure

Modifiez les canaux CLUSRCVR existants sur TOKYO.

Pour indiquer que le canal A du réseau est le canal principal et que le canal B du réseau est le canal secondaire, utilisez les commandes suivantes:

- a) ALTER CHANNEL(INVENTORY.TOKYO.NETA) CHLTYPE(CLUSRCVR) NETPRTY(2) DESCR('Main cluster-receiver channel for TOKYO')
- b) ALTER CHANNEL(INVENTORY.TOKYO.NETB) CHLTYPE(CLUSRCVR) NETPRTY(1) DESCR('Backup cluster-receiver channel for TOKYO')

Que faire ensuite

En configurant le canal avec des priorités de réseau différentes, vous avez défini sur le cluster que vous disposez d'un réseau principal et d'un réseau secondaire. Les gestionnaires de files d'attente du cluster qui utilisent ces canaux utilisent automatiquement le réseau principal chaque fois qu'il est disponible. Le basculement des gestionnaires de files d'attente pour utiliser le réseau secondaire lorsque le réseau principal n'est pas disponible.

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans «Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343. Il contient trois gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets, PARIS un référentiel partiel. L'application d'inventaire s'exécute sur le système à New York, connecté au gestionnaire de files d'attente NEWYORK . L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ .
- Un nouveau magasin est en cours de création à Chicago pour fournir une sauvegarde du système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour ajouter une file d'attente qui servira de sauvegarde.

Procédure

1. Choisissez le référentiel complet auquel CHICAGO fait référence en premier.

Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets pour collecter des informations sur le cluster. Il crée son propre référentiel partiel.

Le référentiel que vous choisissez pour un gestionnaire de files d'attente particulier n'a aucune importance particulière. Dans cet exemple, NEWYORK est choisi. Une fois que le nouveau gestionnaire de files d'attente a rejoint le cluster, il communique avec les deux référentiels.

2. Définissez le canal CLUSRCVR .

Chaque gestionnaire de files d'attente d'un cluster doit définir un récepteur de cluster sur lequel il peut recevoir des messages. Sous CHICAGO, définissez:

```
DEFINE CHANNEL (INVENTORY.CHICAGO) CHLTYPE (CLUSRCVR) TRPTYPE (TCP)
CONNAME (CHICAGO.CMSTORE.COM) CLUSTER (INVENTORY) DESCR ('Cluster-receiver
channel for CHICAGO')
```

3. Définissez un canal CLUSSDR sur le gestionnaire de files d'attente CHICAGO.

Chaque gestionnaire de files d'attente d'un cluster doit définir un canal émetteur de cluster sur lequel il peut envoyer des messages à son premier référentiel complet. Dans ce cas, nous avons choisi NEWYORK, de sorte que CHICAGO a besoin de la définition suivante:

```
DEFINE CHANNEL (INVENTORY.NEWYORK) CHLTYPE (CLUSSDR) TRPTYPE (TCP)
CONNAME (NEWYORK.CHSTORE.COM) CLUSTER (INVENTORY) DESCR ('Cluster-sender
channel from CHICAGO to repository at NEWYORK')
```

4. Modifiez la file d'attente de cluster existante INVENTQ.

Le INVENTQ qui est déjà hébergé par le gestionnaire de files d'attente NEWYORK est l'instance principale de la file d'attente.

```
ALTER QLOCAL (INVENTQ) CLWLPRTY (2)
```

5. Recherchez les affinités de message dans l'application d'inventaire.


Avant de continuer, assurez-vous que l'application d'inventaire ne possède pas de dépendances sur la séquence de traitement des messages.

6. Installez l'application d'inventaire sur le système dans CHICAGO.

7. Définition de la file d'attente de cluster de secours INVENTQ

Le INVENTQ qui est déjà hébergé par le gestionnaire de files d'attente NEWYORK doit également être hébergé en tant que sauvegarde par CHICAGO. Définissez-le sur le gestionnaire de files d'attente CHICAGO comme suit:

```
DEFINE QLOCAL (INVENTQ) CLUSTER (INVENTORY) CLWLPRTY (1)
```

 Maintenant que vous avez terminé toutes les définitions, si ce n'est pas déjà fait, démarrez l'initiateur de canal sur IBM MQ for z/OS.

Sur toutes les plateformes, démarrez un programme d'écoute sur le gestionnaire de files d'attente CHICAGO. Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Résultats

La [Figure 62](#), à la page 435 présente le cluster configuré par cette tâche.

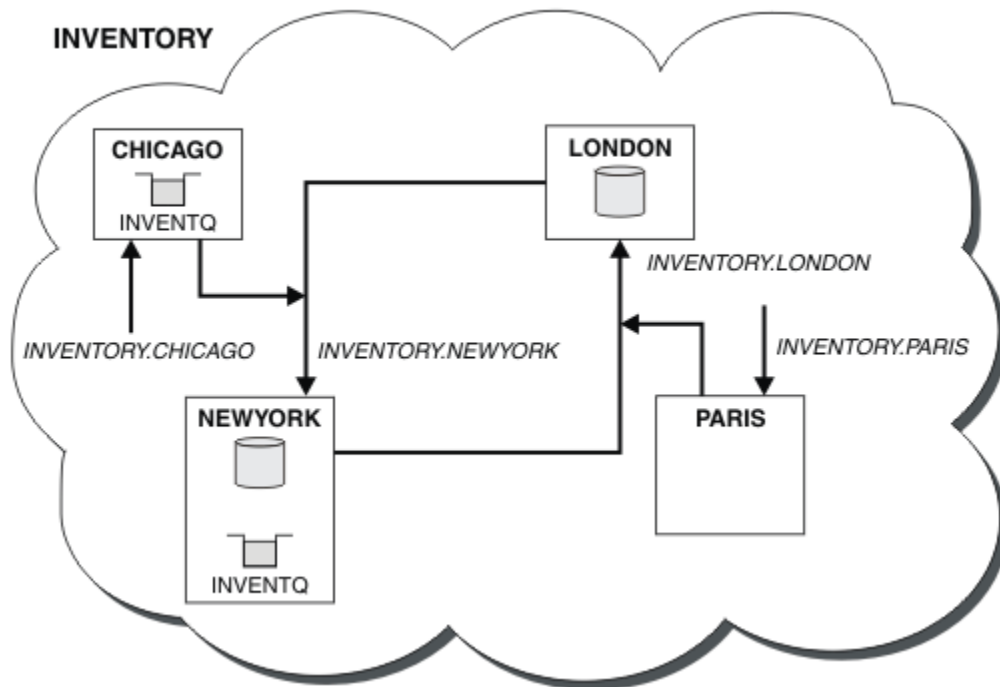


Figure 62. Le cluster INVENTORY, avec quatre gestionnaires de files d'attente

La file d'attente INVENTQ et l'application d'inventaire sont désormais hébergées sur deux gestionnaires de files d'attente du cluster. Le gestionnaire de files d'attente CHICAGO est une sauvegarde. Les messages placés dans INVENTQ sont acheminés vers NEWYORK sauf s'ils sont indisponibles lorsqu'ils sont envoyés à la place à CHICAGO.

Remarque :

La disponibilité d'un gestionnaire de files d'attente éloignées dépend du statut du canal de ce gestionnaire de files d'attente. Lorsque les canaux démarrent, leur état change plusieurs fois, certains d'entre eux étant moins préférables à l'algorithme de gestion de la charge de travail du cluster. En pratique, cela signifie que les destinations de priorité inférieure (sauvegarde) peuvent être choisies alors que les canaux vers les destinations de priorité supérieure (principale) sont en cours de démarrage.

Si vous devez vous assurer qu'aucun message n'est acheminé vers une destination de sauvegarde, n'utilisez pas CLWLPRTY. Envisagez d'utiliser des files d'attente distinctes ou CLWLRANK avec un basculement manuel du serveur principal vers le serveur de sauvegarde.

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Une application de vérification des prix doit être installée sur différents gestionnaires de files d'attente. Pour limiter le nombre de canaux utilisés à un nombre faible, le nombre de canaux actifs exécutés par chaque serveur est limité. L'application est gérée par l'arrivée de messages dans la file d'attente PRICEQ.
- Quatre gestionnaires de files d'attente de serveur hébergent l'application de vérification des prix. Deux gestionnaires de files d'attente de requêtes envoient des messages à PRICEQ pour demander un prix. Deux autres gestionnaires de files d'attente sont configurés en tant que référentiels complets.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour limiter le nombre de canaux utilisés.

Procédure

1. Choisissez deux référentiels complets.

Choisissez deux gestionnaires de files d'attente comme référentiels complets pour votre cluster de vérification des prix. Ils sont appelés REPOS1 et REPOS2.

Entrez la commande suivante :

```
ALTER QMGR REPOS(PRICECHECK)
```

2. Définissez un canal CLUSRCVR sur chaque gestionnaire de files d'attente.

Sur chaque gestionnaire de files d'attente du cluster, définissez un canal récepteur de cluster et un canal émetteur de cluster. Peu importe ce qui est défini en premier.

```
DEFINE CHANNEL(PRICECHECK.SERVE1) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)  
CONNNAME(SERVER1.COM) CLUSTER(PRICECHECK) DESCR('Cluster-receiver channel')
```

3. Définissez un canal CLUSSDR sur chaque gestionnaire de files d'attente.

Créez une définition CLUSSDR sur chaque gestionnaire de files d'attente pour lier ce gestionnaire de files d'attente à l'un ou l'autre des gestionnaires de files d'attente de référentiel complet.

```
DEFINE CHANNEL (PRICECHECK.REPOS1) CHLTYPE (CLUSSDR) TRPTYPE (TCP)
CONNAME (REPOS1.COM) CLUSTER (PRICECHECK) DESCR ('Cluster-sender channel to
repository queue manager')
```

4. Installez l'application de vérification des prix.
5. Définissez la file d'attente PRICEQ sur tous les gestionnaires de files d'attente du serveur.

Exécutez la commande suivante sur chacun d'eux:

```
DEFINE QLOCAL (PRICEQ) CLUSTER (PRICECHECK)
```

6. Restreindre le nombre de canaux utilisés par les requêtes

Sur les gestionnaires de files d'attente de requêtes, nous limitons le nombre de canaux actifs utilisés en émettant les commandes suivantes sur chacun d'eux:

```
ALTER QMGR CLWLMRUC (2)
```

7. Démarrez un programme d'écoute.

z/OS Si ce n'est pas déjà fait, démarrez l'initiateur de canal sur IBM MQ for z/OS.

Sur toutes les plateformes, démarrez un programme d'écoute. Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Résultats

La [Figure 63](#), à la page 437 présente le cluster configuré par cette tâche.

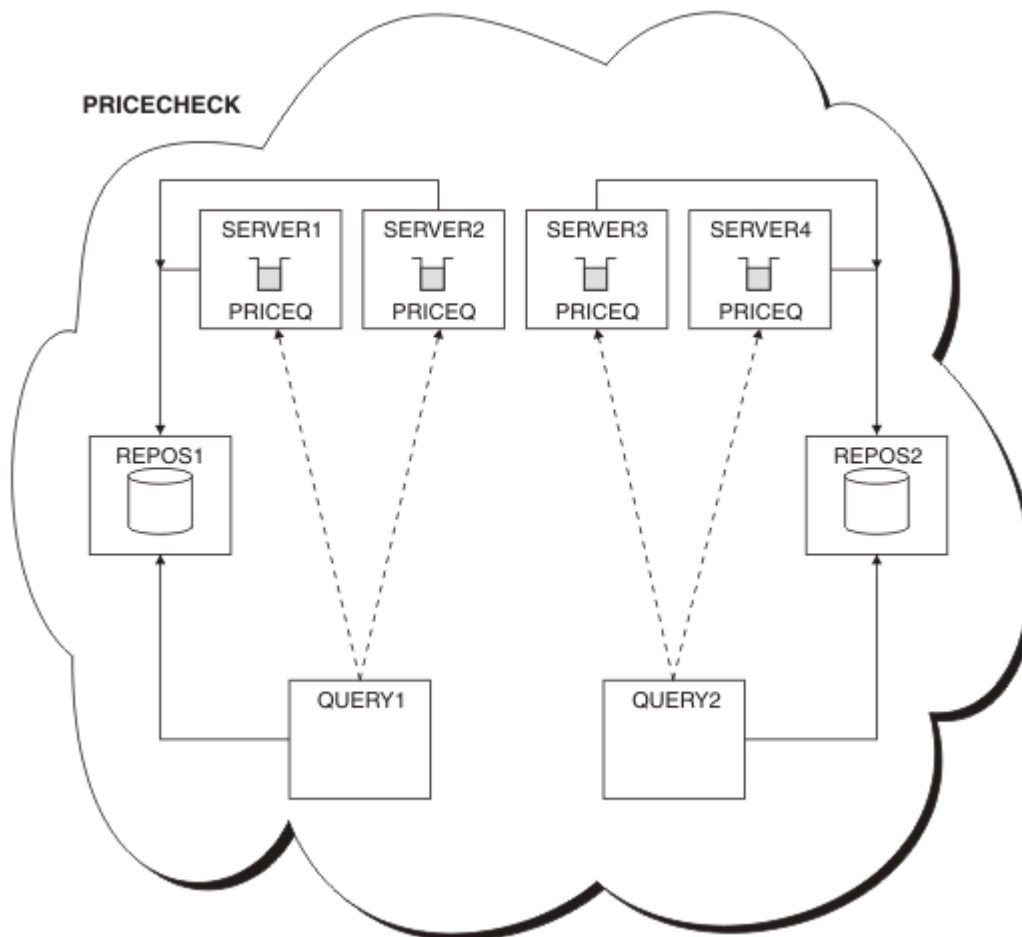


Figure 63. Le cluster PRICECHECK, avec quatre gestionnaires de files d'attente de serveur, deux référentiels et deux gestionnaires de files d'attente de requêtes

Bien que quatre instances de la file d'attente PRICEQ soient disponibles dans le cluster PRICECHECK , chaque gestionnaire de files d'attente de requête n'en utilise que deux sur deux. Par exemple, le gestionnaire de files d'attente QUERY1 ne comporte que des canaux actifs vers les gestionnaires de files d'attente SERVER1 et SERVER2 . Si SERVER1 devient indisponible, le gestionnaire de files d'attente QUERY1 commence alors à utiliser un autre gestionnaire de files d'attente, par exemple SERVER3.

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Avant de commencer

Remarque : Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans «Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343. Il contient trois gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets, PARIS un référentiel partiel et insère des messages à partir de INVENTQ. L'application d'inventaire s'exécute sur le système de New York connecté au gestionnaire de files d'attente NEWYORK . L'application est gérée par l'arrivée de messages dans la file d'attente INVENTQ .
- Un nouveau magasin est en train d'être créé à Los Angeles. Pour fournir une capacité supplémentaire, vous souhaitez exécuter le système d'inventaire à Los Angeles ainsi qu'à New York. Le nouveau gestionnaire de files d'attente peut traiter deux fois plus de messages que New York.

Pourquoi et quand exécuter cette tâche

Procédez comme suit pour ajouter un gestionnaire de files d'attente plus puissant qui héberge une file d'attente.

Procédure

1. Choisissez le référentiel complet auquel LOSANGELES fait référence en premier.
2. Chaque gestionnaire de files d'attente d'un cluster doit faire référence à l'un ou l'autre des référentiels complets pour collecter des informations sur le cluster. Il crée son propre référentiel partiel. Il n'a pas de signification particulière pour le référentiel que vous choisissez. Dans cet exemple, NEWYORK est choisi. Une fois que le nouveau gestionnaire de files d'attente a rejoint le cluster, il communique avec les deux référentiels.

```
DEFINE CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSSDR) TRPTYPE(TCP)
CONNAME(NEWYORK.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-sender channel from LOSANGELES to repository at NEWYORK')
```

3. Définissez le canal CLUSRCVR sur le gestionnaire de files d'attente LOSANGELES.
Chaque gestionnaire de files d'attente d'un cluster doit définir un canal récepteur de cluster sur lequel il peut recevoir des messages. Sous LOSANGELES, définissez:

```
DEFINE CHANNEL(INVENTORY.LOSANGELES) CHLTYPE(CLUSRCVR) TRPTYPE(TCP)
CONNAME(LOSANGELES.CHSTORE.COM) CLUSTER(INVENTORY)
DESCR('Cluster-receiver channel for queue manager LOSANGELES')
CLWLWGHT(2)
```


Le canal récepteur de cluster annonce la disponibilité du gestionnaire de files d'attente pour recevoir les messages des autres gestionnaires de files d'attente du cluster INVENTORY. La définition de CLWLWGHT sur deux garantit que le gestionnaire de files d'attente de Los Angeles reçoit deux fois plus de messages d'inventaire que New York (lorsque le canal pour NEWYORK est défini sur un).

4. Modifiez le canal CLUSRCVR sur le gestionnaire de files d'attente NEWYORK.
Assurez-vous que le gestionnaire de files d'attente de Los Angeles reçoit deux fois plus de messages d'inventaire que New York. Modifiez la définition du canal récepteur de cluster.

```
ALTER CHANNEL(INVENTORY.NEWYORK) CHLTYPE(CLUSRCVR) CLWLWGHT(1)
```
5. Recherchez les affinités de message dans l'application d'inventaire.
Avant de continuer, assurez-vous que l'application d'inventaire ne possède pas de dépendances sur la séquence de traitement des messages.
6. Installation de l'application d'inventaire sur le système à Los Angeles
7. Définissez la file d'attente de cluster INVENTQ.

La file d'attente INVENTQ, qui est déjà hébergée par le gestionnaire de files d'attente NEWYORK, doit également être hébergée par LOSANGELES. Définissez-le sur le gestionnaire de files d'attente LOSANGELES comme suit:

```
DEFINE QLOCAL(INVENTQ) CLUSTER(INVENTORY)
```

 Maintenant que vous avez terminé toutes les définitions, si ce n'est pas déjà fait, démarrez l'initiateur de canal sur IBM MQ for z/OS.

Sur toutes les plateformes, démarrez un programme d'écoute sur le gestionnaire de files d'attente LOSANGELES. Le programme d'écoute écoute les demandes réseau entrantes et démarre le canal récepteur de cluster lorsqu'il est nécessaire.

Résultats

La «Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente», à la page 438 présente le cluster configuré par cette tâche.

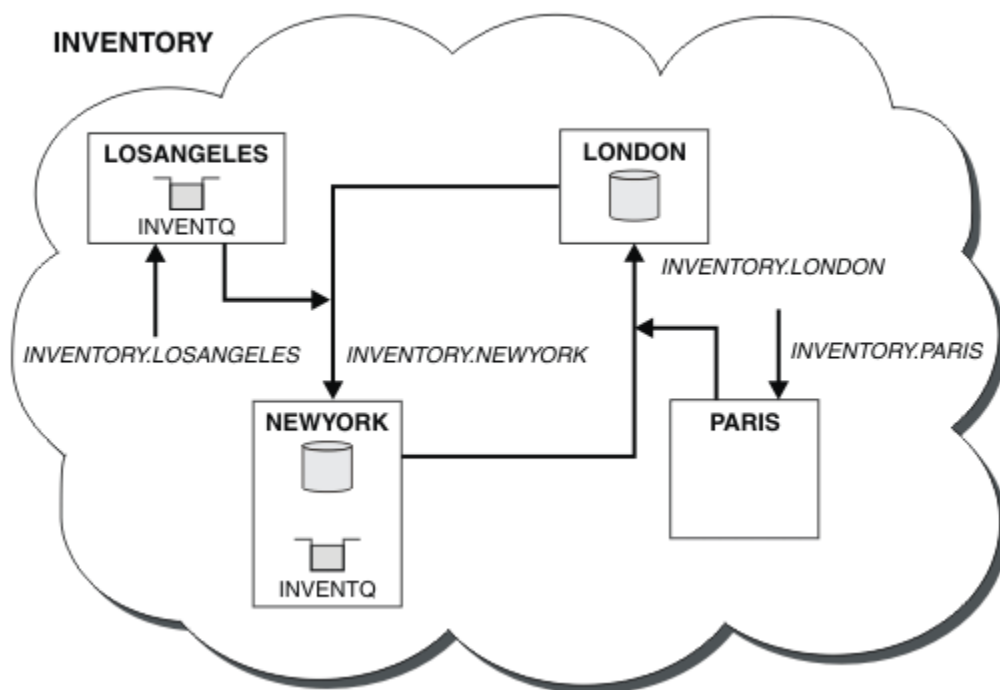


Figure 64. Cluster INVENTORY avec quatre gestionnaires de files d'attente

Cette modification du cluster a été effectuée sans que vous ayez à modifier les gestionnaires de files d'attente LONDON et PARIS. Les référentiels de ces gestionnaires de files d'attente sont mis à jour automatiquement avec les informations dont ils ont besoin pour pouvoir envoyer des messages à INVENTQ à l'adresse LOSANGELES.

Que faire ensuite

La file d'attente INVENTQ et l'application d'inventaire sont hébergées sur deux gestionnaires de files d'attente dans le cluster. La configuration augmente leur disponibilité, accélère le débit des messages et permet de répartir la charge de travail entre les deux gestionnaires de files d'attente. Les messages insérés dans INVENTQ par LOSANGELES ou NEWYORK sont traités par l'instance sur le gestionnaire de files d'attente local dans la mesure du possible. Les messages placés par LONDON ou PARIS sont acheminés vers LOSANGELES ou NEWYORK, deux fois plus de messages étant envoyés à LOSANGELES.

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Tâches associées

[Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local](#)

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Programmation d'application et clusters

Il n'est pas nécessaire d'apporter des modifications à la programmation pour tirer parti de plusieurs instances de la même file d'attente. Toutefois, certains programmes ne fonctionnent pas correctement, sauf si une séquence de messages est envoyée à la même instance d'une file d'attente.

Les applications peuvent ouvrir une file d'attente à l'aide de l'appel MQOPEN . Les applications utilisent l'appel MQPUT pour placer des messages dans une file d'attente ouverte. Les applications peuvent placer un message unique dans une file d'attente qui n'est pas déjà ouverte, à l'aide de l'appel MQPUT1 .

Si vous configurez des clusters ayant plusieurs instances de la même file d'attente, il n'y a pas de considérations spécifiques en matière de programmation d'application. Toutefois, pour tirer parti des aspects de la gestion de la charge de travail liés à la mise en cluster, vous devrez peut-être modifier vos applications. Si vous configurez un réseau dans lequel il existe plusieurs définitions de la même file d'attente, recherchez les affinités de message dans vos applications.

Supposons, par exemple, que vous disposiez de deux applications qui s'appuient sur une série de messages circulant entre elles sous la forme de questions et de réponses. Vous souhaitez probablement que les réponses soient renvoyées au gestionnaire de files d'attente qui a envoyé une question. Il est important que la routine de gestion de la charge de travail n'envoie pas les messages à un gestionnaire de files d'attente qui héberge une copie de la file d'attente de réponses.

Vous pouvez avoir des applications qui nécessitent que les messages soient traités en séquence (par exemple, une application de réplication de base de données qui envoie des lots de messages qui doivent être extraits en séquence). L'utilisation de messages segmentés peut également entraîner un problème d'affinité.

Ouverture d'une version locale ou distante de la file d'attente cible

Tenez compte de la façon dont le gestionnaire de files d'attente choisit d'utiliser une version locale ou distante de la file d'attente cible.

1. Le gestionnaire de files d'attente ouvre la version locale de la file d'attente cible pour lire les messages ou pour définir les attributs de la file d'attente.
2. Le gestionnaire de files d'attente ouvre une instance de la file d'attente cible dans laquelle écrire des messages, si au moins l'une des conditions suivantes est vérifiée:
 - Il n'existe pas de version locale de la file d'attente cible.
 - Le gestionnaire de files d'attente spécifie CLWLUSEQ (ANY) sous ALTER QMGR.
 - La file d'attente du gestionnaire de files d'attente indique CLWLUSEQ (ANY) .

Concepts associés

Exemple de cluster avec plusieurs instances d'une file d'attente

Dans cet exemple de cluster avec plusieurs instances d'une file d'attente, les messages sont acheminés vers différentes instances de la file d'attente. Vous pouvez forcer l'envoi d'un message à une instance spécifique de la file d'attente et choisir d'envoyer une séquence de messages à l'un des gestionnaires de files d'attente.

Tâches associées

Ajout d'un gestionnaire de files d'attente qui héberge une file d'attente en local

Suivez ces instructions pour ajouter une instance de INVENTQ afin de fournir une capacité supplémentaire pour exécuter le système d'application d'inventaire à Paris et à New York.

Utilisation de deux réseaux dans un cluster

Suivez ces instructions pour ajouter un nouveau magasin dans TOKYO où il existe deux réseaux différents. Les deux doivent être disponibles pour pouvoir communiquer avec le gestionnaire de files d'attente à Tokyo.

Utilisation d'un réseau principal et d'un réseau secondaire dans un cluster

Suivez ces instructions pour faire d'un réseau le réseau principal et d'un autre le réseau de secours. Utilisez le réseau de secours en cas de problème avec le réseau principal.

Ajout d'une file d'attente pour agir en tant que sauvegarde

Suivez ces instructions pour fournir une sauvegarde à Chicago pour le système d'inventaire qui s'exécute maintenant à New York. Le système de Chicago n'est utilisé que lorsqu'il y a un problème avec le système de New York.

Restriction du nombre de canaux utilisés

Suivez ces instructions pour limiter le nombre de canaux actifs que chaque serveur exécute lorsqu'une application de vérification des prix est installée sur différents gestionnaires de files d'attente.

Ajout d'un gestionnaire de files d'attente plus puissant qui héberge une file d'attente

Suivez ces instructions pour fournir une capacité supplémentaire en exécutant le système d'inventaire à Los Angeles ainsi qu'à New York, où Los Angeles peut traiter deux fois plus de messages que New York.

Traitement des affinités de message

Les affinités de messages font rarement partie d'une bonne conception de la programmation. Vous devez supprimer les affinités de message pour utiliser complètement la mise en cluster. Si vous ne pouvez pas supprimer les affinités de message, vous pouvez forcer la distribution des messages associés à l'aide du même canal et au même gestionnaire de files d'attente.

Si vous avez des applications avec des affinités de message, supprimez les affinités avant de commencer à utiliser des clusters.

La suppression des affinités de message améliore la disponibilité des applications. Une application envoie un lot de messages ayant des affinités de message à un gestionnaire de files d'attente. Le gestionnaire de files d'attente échoue après n'avoir reçu qu'une partie du lot. Le gestionnaire de files d'attente émetteur doit attendre qu'il récupère et traite le lot de messages incomplet avant de pouvoir envoyer d'autres messages.

La suppression des affinités de messages améliore également l'évolutivité des applications. Un lot de messages avec des affinités peut verrouiller les ressources sur le gestionnaire de files d'attente de destination en attendant les messages suivants. Ces ressources peuvent rester verrouillées pendant de longues périodes, ce qui empêche d'autres applications d'effectuer leur travail.

En outre, les affinités de messages empêchent les routines de gestion de la charge de travail du cluster de faire le meilleur choix du gestionnaire de files d'attente.

Pour supprimer les affinités, tenez compte des possibilités suivantes:

- Transmission des informations d'état dans les messages
- Gestion des informations d'état dans la mémoire rémanente accessible à n'importe quel gestionnaire de files d'attente, par exemple dans une base de données Db2

- Réplication des données en lecture seule afin qu'elles soient accessibles à plusieurs gestionnaires de files d'attente

S'il n'est pas approprié de modifier vos applications pour supprimer les affinités de message, il existe un certain nombre de solutions possibles au problème.

Nommez une destination spécifique dans l'appel MQOPEN

Indiquez le nom de la file d'attente éloignée et le nom du gestionnaire de files d'attente sur chaque appel MQOPEN , et tous les messages insérés dans la file d'attente à l'aide de cet identificateur d'objet sont placés dans le même gestionnaire de files d'attente, qui peut être le gestionnaire de files d'attente local.

La spécification du nom de la file d'attente éloignée et du nom du gestionnaire de files d'attente sur chaque appel MQOPEN présente des inconvénients:

- Aucun équilibrage de charge n'est effectué. Vous ne tirez pas parti des avantages de l'équilibrage de charge de cluster.
- Si le gestionnaire de files d'attente cible est distant et qu'il comporte plusieurs canaux, les messages peuvent emprunter des routes différentes et la séquence de messages n'est toujours pas conservée.
- Si votre gestionnaire de files d'attente possède une définition pour une file d'attente de transmission portant le même nom que le gestionnaire de files d'attente de destination, les messages sont placés dans cette file d'attente de transmission plutôt que dans la file d'attente de transmission du cluster.

Renvoie le nom du gestionnaire de files d'attente dans la zone du gestionnaire de files d'attente de réponse

Autorisez le gestionnaire de files d'attente qui reçoit le premier message d'un lot à renvoyer son nom dans sa réponse. Pour ce faire, il utilise la zone ReplyToQMGR du descripteur de message. Le gestionnaire de files d'attente à l'extrémité émettrice peut ensuite extraire le nom du gestionnaire de files d'attente de réponse et le spécifier sur tous les messages suivants.

L'utilisation des informations ReplyToQMGR dans la réponse présente les inconvénients suivants:

- Le gestionnaire de files d'attente demandeur doit attendre une réponse à son premier message
- Vous devez écrire du code supplémentaire pour rechercher et utiliser les informations ReplyToQMGR avant d'envoyer les messages suivants
- S'il existe plusieurs routes vers le gestionnaire de files d'attente, la séquence des messages risque de ne pas être conservée

Définissez l'option MQ00_BIND_ON_OPEN sur l'appel MQOPEN

Forcez l'insertion de tous vos messages dans la même destination à l'aide de l'option MQ00_BIND_ON_OPEN de l'appel MQOPEN . MQ00_BIND_ON_OPEN ou MQ00_BIND_ON_GROUP doit être spécifié lors de l'utilisation de [groupes de messages](#) avec des clusters pour garantir que tous les messages du groupe sont traités à la même destination.

En ouvrant une file d'attente et en spécifiant MQ00_BIND_ON_OPEN, vous forcez l'envoi de tous les messages envoyés à cette file d'attente à la même instance de la file d'attente. MQ00_BIND_ON_OPEN lie tous les messages au même gestionnaire de files d'attente et à la même route. Par exemple, s'il existe une route IP et une route NetBIOS vers la même destination, l'une d'elles est sélectionnée lorsque la file d'attente est ouverte et cette sélection est prise en compte pour tous les messages insérés dans la même file d'attente à l'aide de l'identificateur d'objet obtenu.

En spécifiant MQ00_BIND_ON_OPEN , vous forcez l'acheminement de tous les messages vers la même destination. Par conséquent, les applications avec des affinités de message ne sont pas interrompues. Si la destination n'est pas disponible, les messages restent dans la file d'attente de transmission jusqu'à ce qu'elle redevienne disponible.

MQ00_BIND_ON_OPEN s'applique également lorsque le nom du gestionnaire de files d'attente est spécifié dans le descripteur d'objet lorsque vous ouvrez une file d'attente. Il peut y avoir plusieurs routes vers le gestionnaire de files d'attente nommé. Par exemple, il peut y avoir plusieurs chemins réseau ou un

autre gestionnaire de files d'attente peut avoir défini un alias. Si vous spécifiez MQ00_BIND_ON_OPEN, une route est sélectionnée lorsque la file d'attente est ouverte.

Remarque : Il s'agit de la technique recommandée. Cependant, il ne fonctionne pas dans une configuration à plusieurs tronçons dans laquelle un gestionnaire de files d'attente annonce un alias pour une file d'attente de cluster. Elle n'est pas non plus utile dans les situations où les applications utilisent des files d'attente différentes sur le même gestionnaire de files d'attente pour différents groupes de messages.

Une alternative à la spécification de MQ00_BIND_ON_OPEN dans l'appel MQOPEN consiste à modifier vos définitions de file d'attente. Dans vos définitions de file d'attente, spécifiez DEFBIND(OPEN) et autorisez l'option DefBind dans l'appel MQOPEN à la valeur par défaut MQ00_BIND_AS_Q_DEF.

Définissez l'option MQ00_BIND_ON_GROUP sur l'appel MQOPEN

Forcez l'insertion de tous les messages d'un groupe dans la même destination à l'aide de l'option MQ00_BIND_ON_GROUP de l'appel MQOPEN. MQ00_BIND_ON_OPEN ou MQ00_BIND_ON_GROUP doit être spécifié lors de l'utilisation de groupes de messages avec des clusters pour garantir que tous les messages du groupe sont traités à la même destination.

En ouvrant une file d'attente et en spécifiant MQ00_BIND_ON_GROUP, vous forcez l'envoi de tous les messages d'un groupe envoyés à cette file d'attente à la même instance de la file d'attente. MQ00_BIND_ON_GROUP lie tous les messages d'un groupe au même gestionnaire de files d'attente et à la même route. Par exemple, s'il existe une route IP et une route NetBIOS vers la même destination, l'une d'elles est sélectionnée lorsque la file d'attente est ouverte et cette sélection est respectée pour tous les messages d'un groupe placés dans la même file d'attente à l'aide de l'identificateur d'objet obtenu.

En spécifiant MQ00_BIND_ON_GROUP, vous forcez tous les messages d'un groupe à être acheminés vers la même destination. Par conséquent, les applications avec des affinités de message ne sont pas interrompues. Si la destination n'est pas disponible, les messages restent dans la file d'attente de transmission jusqu'à ce qu'elle redevienne disponible.

MQ00_BIND_ON_GROUP s'applique également lorsque le nom du gestionnaire de files d'attente est spécifié dans le descripteur d'objet lorsque vous ouvrez une file d'attente. Il peut y avoir plusieurs routes vers le gestionnaire de files d'attente nommé. Par exemple, il peut y avoir plusieurs chemins réseau ou un autre gestionnaire de files d'attente peut avoir défini un alias. Si vous spécifiez MQ00_BIND_ON_GROUP, une route est sélectionnée lorsque la file d'attente est ouverte.

Pour que MQ00_BIND_ON_GROUP soit effectif, vous devez inclure l'option d'insertion MQPMO_LOGICAL_ORDER dans MQPUT. Vous pouvez définir **GroupId** dans le MQMD du message sur MQGI_NONE et vous devez inclure les indicateurs de message suivants dans la zone MQMD **MsgFlags** des messages:

- Dernier message du groupe: MQMF_LAST_MSG_IN_GROUP
- Tous les autres messages du groupe: MQMF_MSG_IN_GROUP

Si MQ00_BIND_ON_GROUP est spécifié mais que les messages ne sont pas regroupés, le comportement est équivalent à MQ00_BIND_NOT_FIXED.

Remarque : Il s'agit de la technique recommandée pour s'assurer que les messages d'un groupe sont envoyés à la même destination. Cependant, il ne fonctionne pas dans une configuration à plusieurs tronçons dans laquelle un gestionnaire de files d'attente annonce un alias pour une file d'attente de cluster.

Une alternative à la spécification de MQ00_BIND_ON_GROUP dans l'appel MQOPEN consiste à modifier vos définitions de file d'attente. Dans vos définitions de file d'attente, spécifiez DEFBIND(GROUP) et autorisez l'option DefBind dans l'appel MQOPEN à la valeur par défaut MQ00_BIND_AS_Q_DEF.

Écriture d'un programme d'exit de charge de travail de cluster personnalisé

Au lieu de modifier vos applications, vous pouvez contourner le problème d'affinités de message en écrivant un programme d'exit de charge de travail de cluster. L'écriture d'un programme d'exit de charge

de travail de cluster n'est pas facile et n'est pas une solution recommandée. Le programme devrait être conçu pour reconnaître l'affinité en inspectant le contenu des messages. Une fois l'affinité reconnue, le programme doit forcer l'utilitaire de gestion de charge de travail à acheminer tous les messages associés vers le même gestionnaire de files d'attente.

Multi Configuration d'un cluster uniforme

Les clusters uniformes permettent aux applications d'être conçues pour l'échelle et la disponibilité, et peuvent se connecter à n'importe quel gestionnaire de files d'attente au sein de ce cluster uniforme.

Avant de commencer

Pour une introduction à la mise en cluster, voir [Clusters](#). Pour une introduction aux clusters uniformes, voir «A propos des clusters uniformes», à la page 445.

Pourquoi et quand exécuter cette tâche

Les clusters uniformes utilisent la mise en cluster IBM MQ pour la communication entre les gestionnaires de files d'attente et l'équilibrage de la charge de travail entre les files d'attente. Toutefois, ils diffèrent des clusters IBM MQ classiques des manières suivantes:


- Les clusters uniformes comportent généralement un plus petit nombre de gestionnaires de files d'attente dans le cluster. Vous ne devez pas créer un cluster uniforme avec plus de 10 gestionnaires de files d'attente.
- Chaque membre du cluster possède une configuration quasi identique.
- Le cluster est généralement utilisé par une seule application ou un groupe d'applications associées.
- Le nombre d'instances d'application se connectant au cluster doit être supérieur ou égal au nombre de gestionnaires de files d'attente.

Vous pouvez simplifier la création d'un cluster uniforme, puis conserver la configuration identique entre les membres du cluster uniforme, à l'aide de la configuration automatique et de la prise en charge de la mise en cluster automatique.

Procédure

- [En savoir plus sur les clusters uniformes](#)
- [Création d'un cluster uniforme](#)
- [Création d'un cluster uniforme](#)
- [Interruption d'un gestionnaire de files d'attente d'un cluster uniforme](#)

Multi A propos des clusters uniformes

L'objectif d'un déploiement en cluster uniforme est que les applications puissent être conçues pour l'échelle et la disponibilité et qu'elles puissent se connecter à n'importe quel gestionnaire de files d'attente au sein du cluster uniforme. Cela permet de supprimer toute dépendance sur un gestionnaire de files d'attente spécifique, ce qui améliore la disponibilité et l'équilibrage de la charge de travail du trafic de messagerie.  Les clusters uniformes ne sont pas disponibles sur IBM MQ for z/OS; les groupes de partage de files d'attente fournissent de nombreuses fonctionnalités d'un cluster uniforme.

Les clusters uniformes sont un modèle spécifique d'un cluster IBM MQ qui fournit une petite collection de gestionnaires de files d'attente à haute disponibilité et échelonnée horizontalement. Ces gestionnaires de files d'attente sont configurés de manière presque identique, de sorte qu'une application puisse interagir avec eux en tant que groupe unique. Ainsi, il est plus facile de s'assurer que chaque gestionnaire de files d'attente du cluster est utilisé, en s'assurant automatiquement que les instances d'application sont réparties de manière égale entre les gestionnaires de files d'attente.

Les clusters uniformes suppriment certaines des étapes manuelles qu'un administrateur doit effectuer pour créer et administrer un groupe de gestionnaires de files d'attente indépendants et interconnectés. Ils déplacent une logique de connexion client du client vers le gestionnaire de files d'attente, où les informations sur les niveaux d'activité de l'application peuvent éclairer les décisions des clients quant aux gestionnaires de files d'attente auxquels ils doivent se connecter.

Vous pouvez simplifier la création initiale d'un cluster uniforme, puis conserver la configuration identique entre les membres du cluster uniforme, à l'aide de la configuration automatique et de la prise en charge de la mise en cluster automatique. Lorsque vous utilisez cette fonction, un fichier de configuration décrit le cluster et un autre représente la configuration MQSC à appliquer à tous les gestionnaires de files d'attente du cluster uniforme. A chaque redémarrage du gestionnaire de files d'attente, la configuration est réappliquée et le cluster se forme automatiquement. Pour plus de détails sur l'utilisation de cette fonction, voir [«Création d'un cluster uniforme»](#), à la page 461 .

Pour tirer pleinement parti d'un cluster uniforme, chaque application doit également être mise à l'échelle en plusieurs instances correspondantes, de préférence avec au moins autant d'instances que de gestionnaires de files d'attente, si ce n'est pas le cas.

Un cluster IBM MQ , quelle que soit sa taille, offre plusieurs fonctions:

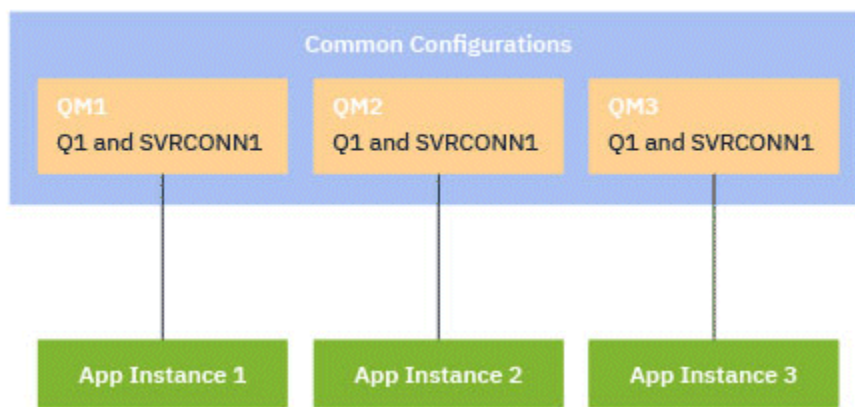
- Répertoire de toutes les ressources de mise en cluster, reconnaissable par n'importe quel membre d'un cluster
- Création automatique de canal et connectivité
- Mise à l'échelle horizontale sur plusieurs files d'attente correspondantes, à l'aide de l'équilibrage de charge de message
- Routage dynamique des messages, en fonction de la disponibilité

Les clusters uniformes utilisent la mise en cluster IBM MQ pour la communication entre les gestionnaires de files d'attente et l'équilibrage de la charge de travail entre les files d'attente. Toutefois, ils diffèrent des clusters IBM MQ classiques des manières suivantes:

- Les clusters uniformes comportent généralement un plus petit nombre de gestionnaires de files d'attente dans le cluster. Vous ne devez pas créer un cluster uniforme avec plus de 10 gestionnaires de files d'attente.
- Chaque membre du cluster possède une configuration quasi identique.
- Le cluster est généralement utilisé par une seule application ou un groupe d'applications associées.
- Le nombre d'instances d'application se connectant au cluster doit être supérieur ou égal au nombre de gestionnaires de files d'attente.

Dans un modèle de cluster uniforme, tous les gestionnaires de files d'attente du cluster offrent les mêmes services de messagerie. Par exemple, vous pouvez configurer tous les membres du cluster pour qu'ils aient les mêmes files d'attente locales définies et permettre aux applications client de se connecter à n'importe quel membre du cluster. Vous pouvez également avoir les mêmes canaux de connexion serveur définis, et éventuellement les mêmes enregistrements de droits d'accès, les mêmes règles d'authentification de canal, etc. Toutefois, les membres du cluster peuvent encore présenter des différences dans les objets et la configuration. Par exemple, certaines applications peuvent créer des files d'attente dynamiques temporaires lorsqu'elles sont connectées à un gestionnaire de files d'attente. En outre, certaines mises à jour de configuration peuvent être déployées sur les membres sur une période donnée ; par exemple, des certificats nouveaux ou mis à jour. Comme avec les clusters IBM MQ standard, deux des gestionnaires de files d'attente nécessitent une configuration supplémentaire pour en faire des gestionnaires de files d'attente de référentiel complet.

Le diagramme suivant montre que les gestionnaires de files d'attente ont des configurations similaires. Ils définissent la même file d'attente appelée Q1 et le même canal de connexion serveur SVRCONN1.



Notez que pour que plusieurs gestionnaires de files d'attente avec des noms de canal de connexion serveur identiques fonctionnent avec une seule table de définition de canal du client (CCDT), vous devez utiliser le format CCDT mis à jour introduit dans IBM MQ 9.1.2. Voir [«Configuration d'une table de définition de canal du client au format JSON»](#), à la page 48.

Noms d'application et instances d'application

Un nom d'application s'affiche en tant qu'attribut `APPLTAG` de la commande **DISPLAY CONN(*) TYPE CONN**. Depuis la IBM MQ 9.1.2, il y a un changement dans la manière dont le nom de l'application est défini.

Une instance d'application est un ensemble de connexions étroitement liées qui fournissent une *unité d'exécution* pour cette application. En règle générale, il s'agit d'un processus de système d'exploitation unique, qui peut comporter un certain nombre d'unités d'exécution et des connexions IBM MQ associées.

Pour plus d'informations sur le nom d'application et les instances d'application, voir [Concepts de développement d'application](#).

Clients reconnectables

Les clients reconnectables peuvent être déplacés afin d'obtenir une distribution de charge de travail uniforme alors que, par définition, un client non reconnectable ne peut pas être reconnecté à un autre gestionnaire de files d'attente. Cependant, il peut encore y avoir une bonne raison de connecter un client non reconnectable à un cluster uniforme: par exemple, parce que le client crée une forme d'état persistant et qu'un autre mécanisme est utilisé pour s'assurer que des instances de l'application s'exécutent dans chacun des gestionnaires de files d'attente.

Applications liées localement

Les clusters uniformes doivent avoir des applications IBM MQ qui se connectent en tant qu'applications client, plutôt que des applications liées localement. Les applications liées localement ne sont pas empêchées de se connecter à des membres de cluster uniformes, mais les clusters uniformes ne peuvent même pas distribuer la charge de travail avec des applications liées localement, car ils ne peuvent pas se connecter à un autre membre du cluster.

Tâches associées

[Spécification du nom d'application dans les langages de programmation pris en charge](#)

Multi **Équilibrage automatique des applications**

L'équilibrage automatique des applications améliore considérablement la distribution et la disponibilité des applications en permettant à un cluster uniforme IBM MQ de gérer étroitement la distribution des applications dans le cluster, plutôt que de s'appuyer sur la randomisation ou sur une épingle manuel des applications à des gestionnaires de files d'attente spécifiques.

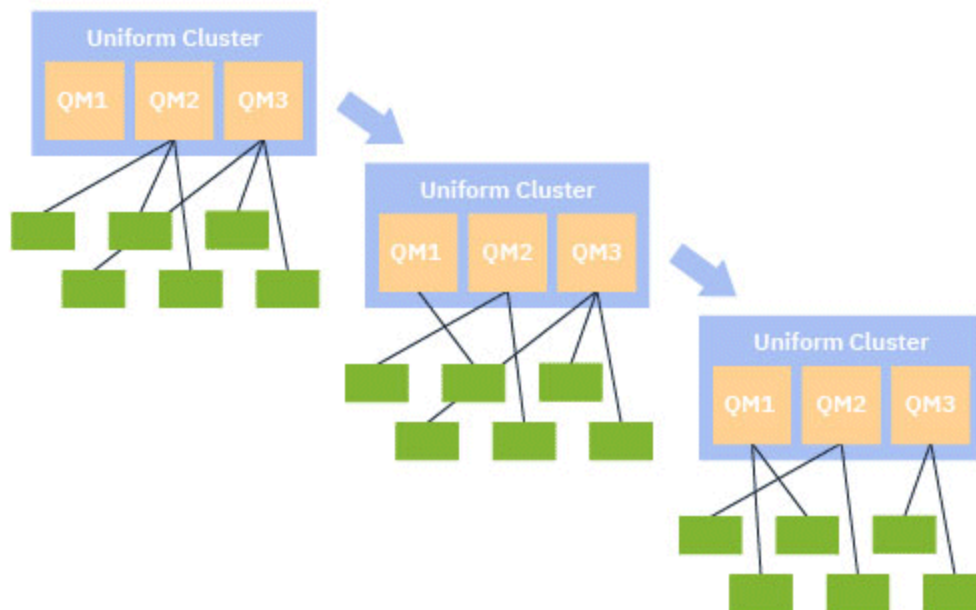
L'équilibrage automatique sur un ensemble de gestionnaires de files d'attente en cluster est pris en charge pour les applications écrites en C, JMS, IBM MQ .NET, XMS .NET.

Lorsqu'il existe au moins autant d'instances de la même application que de gestionnaires de files d'attente, le cluster uniforme garantit en permanence qu'au moins une instance de l'application est connectée à chaque gestionnaire de files d'attente.

Les applications peuvent supprimer une affinité spécifique à un gestionnaire de files d'attente et utiliser à la place une table de définition de canal du client (CCDT) pour rendre aléatoire la connectivité au groupe de gestionnaires de files d'attente dans le cluster uniforme en toute sécurité. Les applications peuvent effectuer cette opération pour les raisons suivantes:

- Lorsqu'il y a suffisamment d'instances d'application consommatrices, il y a toujours une instance des messages de traitement de l'application.
- Lorsque vous arrêtez un gestionnaire de files d'attente, toutes les instances d'application connectées sont réparties équitablement entre les autres gestionnaires de files d'attente du cluster.
- Lorsque vous démarrez un gestionnaire de files d'attente, toutes les instances d'application connectées à d'autres gestionnaires de files d'attente du cluster sont automatiquement rééquilibrées pour inclure le gestionnaire de files d'attente nouvellement démarré.

Cela signifie que le cluster uniforme garantit en permanence une distribution optimale des applications, en maximisant le traitement des messages, même en cas d'indisponibilités planifiées et non planifiées.



Pour atteindre l'équilibrage automatique, les gestionnaires de files d'attente du cluster uniforme partagent périodiquement des informations entre eux. Pour ce faire, ils publient des métadonnées sur les rubriques système sous la branche \$SYS/MQ réservée de l'arborescence de rubriques. Chaque gestionnaire de files d'attente du cluster uniforme s'abonne aux messages publiés par d'autres gestionnaires de files d'attente et génère une image de l'état des applications du cluster uniforme.

Les gestionnaires de files d'attente surveillent la distribution des applications client sur l'ensemble du cluster. Lorsque le nombre d'applications connectées à un gestionnaire de files d'attente spécifique est suffisamment faible pour qu'il détermine que le cluster est déséquilibré, ce gestionnaire de files d'attente publie une demande sur une rubrique système dans l'un des autres gestionnaires de files d'attente du cluster.

Lorsque le message est reçu, le gestionnaire de files d'attente cible demande à l'une de ses applications client de se rediriger vers le gestionnaire de files d'attente demandeur. L'application client reçoit la demande de redirection, ferme sa connexion et se reconnecte au gestionnaire de files d'attente demandeur. Ce mécanisme d'équilibrage automatique est transparent pour l'application. Pour plus d'informations, voir «[Fonctionnement de l'équilibrage automatique](#)», à la page 449.

En distribuant périodiquement des métadonnées sur les applications connectées, le cluster uniforme peut atteindre un rapport largement équilibré entre les applications client et les gestionnaires de files d'attente au fil du temps. Pour éviter que des événements de redirection successifs rapides ne se produisent, l'algorithme d'équilibrage automatique limite le débit auquel les demandes de redirection sont effectuées.

Vous pouvez surveiller l'état en cours des applications dans les gestionnaires de files d'attente d'un cluster et surveiller les instances d'application. Pour plus d'informations, voir [Monitoring application balancing](#). Vous pouvez également résoudre divers problèmes liés à l'équilibrage des applications, comme décrit dans [Traitement des incidents liés à l'équilibrage des applications](#).

Le rééquilibrage n'est utile que pour les applications dont la durée de connexion est longue. Si vous disposez d'applications client avec des temps de connexion courts, par exemple des applications client qui sont écrites pour se connecter et se déconnecter régulièrement à différents gestionnaires de files d'attente, vous devez les configurer comme n'étant pas reconnectables. Cela les retire de l'ensemble d'applications que les gestionnaires de files d'attente tentent d'équilibrer.

Concepts associés

«Comment l'équilibrage automatique utilise la reconnexion automatique», à la page 451

L'équilibrage automatique de cluster uniforme utilise les améliorations apportées à la fonction de reconnexion automatique existante d' IBM MQ.

Multi *Fonctionnement de l'équilibrage automatique*

Dans le cluster uniforme, les connexions client sont regroupées en fonction du nom de l'application. Les applications qui se connectent à un membre du cluster uniforme à l'aide du même nom d'application sont considérées comme équivalentes à toutes les autres applications utilisant le même nom d'application.

L'équilibrage automatique garantit une répartition uniforme des instances d'application entre les membres du cluster. Pour plus d'informations, voir «Noms d'application et instances d'application», à la page 447 . La commande `DISPLAY APSTATUS` permet d'afficher le statut d'une ou de plusieurs applications et instances d'application connectées à un gestionnaire de files d'attente ou à un cluster uniforme.

Par exemple, vous pouvez définir toutes les instances d'une application de demande d'assurance avec le nom d'application "INSURANCE.REQUESTS". Les connexions associées à partir de cette application seront automatiquement regroupées en instances, le cas échéant, et tous les rééquilibrages seront effectués sur une base par instance.

Lorsque de nouvelles instances de l'application se connectent à un membre du cluster uniforme, l'algorithme d'équilibrage automatique évalue les gestionnaires de files d'attente ayant le moins d'instances d' INSURANCE.REQUESTS et redirige certaines connexions vers ces gestionnaires de files d'attente.

L'équilibrage automatique est activé uniquement dans les cas suivants:

- La valeur SHARECNV du canal est supérieure à zéro.
- L'une des conditions suivantes est vraie:
 - L'application client spécifie MQCNO_RECONNECT
 - Le fichier mqcClient.ini spécifie **Defrecon=YES**

Remarque : Les applications avec affinité de gestionnaire de files d'attente-par exemple en raison d'un abonnement durable ou d'une réponse dynamique à la file d'attente-ne peuvent pas être rééquilibrées en toute sécurité et doivent utiliser MQCNO_RECONNECT_QMGR ou aucune option de reconnexion.

Lorsqu'un client est redirigé vers un autre gestionnaire de files d'attente, il utilise, comme d'habitude, les tables de définition de canal du client local (CCDT) pour localiser les informations de connexion pour la nouvelle cible. Il est donc important pour un fonctionnement harmonieux et efficace de l'équilibrage automatique que les clients utilisent une table de définition de canal du client contenant une entrée pour chaque membre du cluster uniforme, ainsi que tout groupe de gestionnaires de files d'attente utilisé pour équilibrer les connexions initiales.

L'utilisation d'une table de définition de canal du client au format JSON simplifie cette opération, car elle autorise plusieurs connexions à l'aide du même nom de connexion serveur. Pour plus d'informations, voir [«Configuration d'une table de définition de canal du client au format JSON»](#), à la page 48.

Concepts associés

[«Comment l'équilibrage automatique utilise la reconnexion automatique»](#), à la page 451

L'équilibrage automatique de cluster uniforme utilise les améliorations apportées à la fonction de reconnexion automatique existante d' IBM MQ.

Equilibrage automatique des applications JMS

Lorsque les applications Jakarta Messaging 3.0 ou Java Message Service 2.0 sont équilibrées automatiquement, les groupes sous-jacents de connexions IBM MQ créés par les applications JMS sont déplacés ensemble.

Depuis la IBM MQ 9.3.0, la propriété **dynamicallyBalanced** est disponible lors de la configuration de ActivationSpecs. Cette propriété indique si un bean géré par message peut être invité à recevoir des messages d'un gestionnaire de files d'attente différent dans le cadre de l'équilibrage des applications dans un cluster uniforme. Pour plus d'informations, voir [Configuration de l'adaptateur de ressources pour les communications entrantes](#).

Pour la gestion des connexions JMS, les clusters uniformes ont le concept d' *instance d'application*. Pour JMS, une *instance d'application* est définie comme une connexion JMS et toutes ses sessions JMS associées.

Une balise de connexion unique est allouée sur la connexion client qui correspond à la connexion JMS, puis la même balise est appliquée aux connexions client qui correspondent aux sessions JMS créées par cette connexion JMS.

Par exemple, si une paire d'applications client exécute des applications JMS sur un cluster uniforme avec un seul gestionnaire de files d'attente actif (gestionnaire de files d'attente 1):

- Le client 1 crée une fabrique de connexions sur laquelle il définit le nom d'application "App1" et crée une connexion JMS et trois sessions JMS. Le client 1 crée quatre connexions client dans le gestionnaire de files d'attente 1, qui partagent chacune la même balise de connexion et qui sont traitées comme une seule instance de "App1".
- Le client 2 crée également une fabrique de connexions sur laquelle il définit le nom d'application "App1" et crée une connexion JMS et deux sessions JMS. Le client 2 crée trois connexions client partageant chacune la même balise de connexion (distincte de celle affectée au client 1), qui est traitée comme une instance unique et distincte de "App1".
- Le gestionnaire de files d'attente voit donc deux instances de "App1"

Lorsque l'équilibrage automatique est effectué, les instances d'application sont déplacées. Un gestionnaire de files d'attente choisit une instance d'application (un groupe de connexions client partageant la même balise de connexion) et demande que l'instance soit déplacée vers un autre gestionnaire de files d'attente. Le code client reçoit la demande et s'assure que toutes les connexions associées (correspondant à une connexion JMS et à ses sessions JMS associées) sont déplacées vers le nouveau gestionnaire de files d'attente.

Par exemple, prenez l'ensemble d'instances d'application précédemment décrit et supposons qu'un nouveau gestionnaire de files d'attente (gestionnaire de files d'attente 2) démarre dans le cluster uniforme.

Le gestionnaire de files d'attente 2 ne fonctionne pas, mais le gestionnaire de files d'attente 1 possède 2 instances de "App1". Par conséquent, le gestionnaire de files d'attente 2 demande que le gestionnaire de files d'attente 1 transfère une instance de "App1" au gestionnaire de files d'attente 2.

Le gestionnaire de files d'attente 1 choisit une instance de "App1" à déplacer. Pour les besoins de l'exemple, supposons qu'il choisisse l'instance créée par le Client 1.

- Le gestionnaire de files d'attente 1 envoie une demande au client 1 pour déplacer son instance de "App1" vers QM2.

- Le client ferme ses quatre connexions client existantes au gestionnaire de files d'attente 1 et crée quatre nouvelles connexions au gestionnaire de files d'attente 2.
- La connexion JMS et ses sessions JMS , à l'exception d'une courte pause dans le traitement, ne doivent normalement pas être perturbées.

Remarque :

Une application peut recevoir une exception JMS si certaines opérations sont en cours au moment où une instance d'application est déplacée.

L'exception JMS comporte une exception IBM MQ liée, à partir de laquelle le code anomalie peut être extrait pour déterminer la cause de l'échec.

Les codes anomalie attendus sont les suivants:

MQRC_CALL_INTERRUPTUE

Cela se produit lorsque, par exemple, un message persistant (valeur par défaut dans JMS) est placé en dehors d'un point de synchronisation, mais que l'opération est interrompue par une reconnexion.

MQRC_BACKED_OUT

Cela se produit lorsque, par exemple, une tentative d'insertion d'un message dans un point de synchronisation est interrompue par une reconnexion.

Concepts associés

«Fonctionnement de l'équilibrage automatique», à la page 449

Dans le cluster uniforme, les connexions client sont regroupées en fonction du nom de l'application. Les applications qui se connectent à un membre du cluster uniforme à l'aide du même nom d'application sont considérées comme équivalentes à toutes les autres applications utilisant le même nom d'application.

«Comment l'équilibrage automatique utilise la reconnexion automatique», à la page 451

L'équilibrage automatique de cluster uniforme utilise les améliorations apportées à la fonction de reconnexion automatique existante d' IBM MQ.

Multi *Comment l'équilibrage automatique utilise la reconnexion automatique*

L'équilibrage automatique de cluster uniforme utilise les améliorations apportées à la fonction de reconnexion automatique existante d' IBM MQ.

Dans les versions de IBM MQ antérieures à IBM MQ 9.2.0, la fonction de reconnexion automatique se reconnecte automatiquement à une instance de secours d'un gestionnaire de files d'attente ou à un autre gestionnaire de files d'attente, en fonction des détails de connexion fournis, généralement une liste de noms de connexion ou une table de définition de canal du client (CCDT).

Dans certains cas, le client IBM MQ effectue la reconnexion en mode silencieux sans que l'application n'en soit consciente. La décision du gestionnaire de files d'attente auquel se reconnecter dépend entièrement de la séquence de noms de connexion dans une liste de noms de connexion ou de la configuration d'équilibrage de charge dans la table de définition de canal du client.

Depuis IBM MQ 9.2.0 , une demande de reconnexion peut être envoyée à un client contenant une indication du gestionnaire de files d'attente auquel le client doit se reconnecter. Dans de nombreux scénarios de reconnexion, tels que l'échec d'un gestionnaire de files d'attente ou l'exécution de la commande **endmqm -r** par l'administrateur, un nom de gestionnaire de files d'attente n'est pas inclus dans les informations de suggestion et le comportement de reconnexion automatique fonctionne comme avant IBM MQ 9.2.0.

Toutefois, si vous avez configuré un cluster uniforme, l'équilibrage automatique des applications envoie périodiquement des demandes de reconnexion aux clients afin d'obtenir un cluster équilibré. Dans ces cas, le cluster uniforme spécifie un nom de gestionnaire de files d'attente dans la suggestion de reconnexion pour s'assurer que les connexions client sont déplacées vers les gestionnaires de files d'attente ayant le moins de connexions.

Pour que l'équilibrage automatique fonctionne, les paramètres suivants doivent être définis:

- Les applications IBM MQ utilisent des tables de définition de canal du client pour extraire les informations de connexion.

- Les cCDT contiennent une entrée pour chaque gestionnaire de files d'attente du cluster uniforme.

Si tel n'est pas le cas, le cluster ne peut pas équilibrer automatiquement les applications entre tous les membres du cluster.

Si une application utilise une version du client IBM MQ antérieure à IBM MQ 9.2.0 et qu'elle est configurée pour prendre en charge la reconnexion automatique du client, elle peut être envoyée par le cluster uniforme pour passer par ses étapes de reconnexion.

Le client ne sera pas invité à se reconnecter à un gestionnaire de files d'attente spécifique, mais passera par la même séquence de logique de reconnexion que celle qu'il ferait pour d'autres événements de reconnexion. Pour obtenir une distribution uniforme des applications client avant IBM MQ 9.2.0 dans le cluster uniforme, assurez-vous que les clients sont configurés pour utiliser des tables de définition de canal du client contenant des entrées pondérées de manière uniforme pour chaque membre du cluster.

Les applications peuvent effectuer plusieurs tentatives de reconnexion avant de se connecter à un gestionnaire de files d'attente qui a besoin de l'instance supplémentaire. Il s'agit donc d'un moyen moins efficace d'obtenir une distribution uniforme des applications dans le cluster. L'équilibrage automatique peut prendre plus de temps dans ces environnements.

Les clients IBM MQ ne prennent pas en charge la reconnexion automatique du client

Si une application utilise une version du client IBM MQ qui ne prend pas en charge la reconnexion automatique du client, elle peut recevoir un code retour d'échec d'un appel MQI.

Si votre application n'a pas été conçue pour gérer les échecs et effectuer des reconnexions manuellement, il peut être nécessaire de désactiver l'équilibrage automatique pour ces applications.

Remarque : L'équilibrage automatique est activé pour toute application identifiée comme étant reconnectable, c'est-à-dire que l'application a MQCNO_RECONNECT dans ses options de connexion effective.

Tâches associées

«Création d'un nouveau cluster uniforme», à la page 461
Comment créer un nouveau cluster uniforme.

Multi *Influence du rééquilibrage des applications dans les clusters uniformes*

Avec l'équilibrage automatique des applications (fonction de clusters uniformes), une connexion d'application peut être demandée pour passer à un autre gestionnaire de files d'attente à tout moment de son cycle de vie.

Introduction

Depuis la IBM MQ 9.3.0, l'algorithme d'équilibrage tente automatiquement de prendre en compte l'état des applications afin de réduire les interruptions du flux d'application. Vous pouvez l'adapter à des applications ou à des instances d'application spécifiques en fournissant à IBM MQ des informations supplémentaires sur le type d'application ou le modèle d'activité IBM MQ exécuté par cette application.

En règle générale, la personne qui développe ou déploie une application client est probablement la mieux placée pour comprendre ce modèle et fournir ces informations au gestionnaire de files d'attente (voir [Déploiement d'applications client flexibles et évolutives](#)), mais elle peut également ou en plus être optimisée par un administrateur.

Notez que si le gestionnaire de files d'attente ne parvient pas à obtenir une distribution uniforme des applications dans un délai raisonnable, les connexions d'application risquent d'être rééquilibrées avec d'autres gestionnaires de files d'attente sans attendre un temps approprié dans leur flux IBM MQ .

Cela peut également être ajusté pour répondre aux exigences. S'il est plus important d'obtenir rapidement une distribution uniforme des applications, vous pouvez configurer le produit pour qu'il attende moins de temps pour trouver un temps approprié pour rééquilibrer une application. Sinon, s'il est plus important d'éviter les interruptions des applications, il est possible de configurer le produit pour qu'il attende toujours un moment pratique pour déplacer l'application.

Pour plus d'informations, voir [Déploiement d'applications client flexibles et évolutives](#) .

Pour les applications .NET, voir [«Influence du rééquilibrage des applications dans .NET»](#), à la page 456 pour plus d'informations.

Pour les applications.XMS.NET , voir [Propriétés de ConnectionFactory](#) pour plus d'informations.

V 9.4.0 Pour les applications JMS, voir [«Influence du rééquilibrage des applications dans IBM MQ classes for JMS»](#), à la page 457 pour plus d'informations.

Comportement d'équilibrage d'application par défaut

Par défaut, l'état de transaction / unité de travail d'une interaction d'application avec un gestionnaire de files d'attente est pris en compte pour toutes les applications.

Pour les transactions locales, l'équilibrage automatique des applications évite d'émettre des demandes de rééquilibrage aux applications qui sont actuellement impliquées dans une transaction. Bien que cela n'élimine pas la possibilité pour une application de recevoir un code retour annulé, car le fait d'atteindre le délai d'attente de rééquilibrage configuré ou une indisponibilité réelle peut toujours entraîner un tel code retour, cela signifie que les applications ne seront généralement pas invitées à se reconnecter alors qu'elles sont au milieu d'une transaction.

Pour les applications qui commencent une nouvelle transaction presque immédiatement après l'exécution de la précédente, il peut y avoir un délai pour l'appel initial dans la nouvelle transaction lors de la fin du rééquilibrage. Cela garantit que l'équilibrage automatique des applications est toujours en mesure d'obtenir une distribution uniforme des applications entre les gestionnaires de files d'attente dans un cluster uniforme.

Si des applications utilisent des transactions à exécution plus longue, vous pouvez envisager d'augmenter la valeur du délai d'attente de rééquilibrage ou de désactiver entièrement cette contrainte. Voir [«Configuration du comportement d'équilibrage»](#), à la page 454 pour des liens sur la façon de contrôler cela dans MQI et .NET, ou Conception d'applications client pour la tolérance aux pannes et l'évolutivité pour le niveau de code équivalent.

Equilibrage demande-réponse

Lorsque le type d'application est défini sur **Request-Reply**, une réponse GET est attendue pour chaque opération PUT exécutée par l'instance d'application. Si l'instance d'application implique plusieurs unités d'exécution ou traite des demandes et des réponses par lots, plusieurs demandes et réponses peuvent être en cours à tout moment.

L'application n'est pas considérée comme éligible pour le déplacement tant que le nombre de demandes envoyées n'est pas égal au nombre de réponses reçues ou que la valeur de backstop du délai d'attente n'est pas dépassée.

Une exception à cette règle est lorsque l'expiration d'un message est configurée pour un message de demande. Il est supposé que les réponses doivent être reçues dans l'intervalle d'expiration du message de demande et que, lorsque tous les messages de demande sont arrivés à expiration, l'algorithme d'équilibrage n'attend plus de réponses supplémentaires avant de considérer que l'instance est éligible pour le déplacement.

Si plusieurs demandes sont en attente, seul le dernier délai d'expiration parmi les messages de demande envoyés est pris en compte. Lorsque des valeurs d'expiration significatives sont utilisées, vous devez configurer le paramètre d'équilibrage **Timeout** pour que l'application ait une valeur au moins aussi élevée que n'importe quelle valeur d'expiration de message envoyé, afin d'éviter de réduire toute fenêtre d'expiration de demande / réponse attendue.

Le modèle précédent convient uniquement aux applications qui s'attendent à avoir des périodes pendant lesquelles il n'y a pas de demandes en attente. Les applications à unités d'exécution multiples complexes, qui envoient et reçoivent constamment des messages, par exemple, risquent de ne jamais pouvoir être rééquilibrées dans ce modèle.

Remarques :

- Aucune tentative n'est effectuée pour corrélérer des demandes et des réponses spécifiques. Par conséquent, si une réponse antérieure dans un lot de messages en cours expire, l'application peut attendre jusqu'à l'expiration de la dernière demande avant d'être éligible à l'équilibrage.
- En particulier, une attention particulière est nécessaire si vous combinez un délai d'expiration illimité et des messages arrivant à expiration, pour des raisons similaires.

Si des messages de demande avec une durée d'expiration limitée sont en attente et que de nouveaux messages sont envoyés avec une durée d'expiration illimitée, le délai d'expiration illimité n'est *pas* pris en compte par l'algorithme d'équilibrage, qui continue à respecter le dernier délai d'expiration en cours.

Dans le cas contraire, des réponses antérieures ayant expiré pourraient empêcher l'application de pouvoir être déplacé. En conséquence, si des réponses de délai d'expiration illimité sont en attente, mais que des demandes arrivant à expiration sont ensuite envoyées, le temps d'attente est réduit à l'expiration la plus longue (limitée).

En règle générale, vous devez éviter qu'une seule instance d'application n'envoie à la fois des messages de demande expirant et n'expirant pas dans une application équilibrée, car l'éligibilité au rééquilibrage devient plus difficile pour un développeur ou un administrateur à suivre ou à définir avec précision.

- Uniquement le délai d'expiration spécifié par l'application émettrice (par exemple, dans l'interface MQI, la valeur de MQMD.**Expiry**) est pris en compte lors de la détermination du délai d'attente des réponses. Les modifications ultérieures apportées à cette valeur, par exemple, l'utilisation de CAPEXPY n'aura pas d'effet sur le temps d'attente.


Configuration du comportement d'équilibrage

Pour influencer précisément le moment où IBM MQ rééquilibre les applications, certains environnements d'application client peuvent fournir des informations au moment de la connexion sur le modèle de messagerie utilisé.

Ces informations sont fournies dans une nouvelle structure appelée *Options d'équilibrage*.

Pour l'interface MQI, voir [«Configuration du comportement d'équilibrage à l'aide de l'interface MQI»](#), à la page 454.

Pour l'équivalent client .NET de cette structure, voir [«Influence du rééquilibrage des applications dans .NET»](#), à la page 456.

 Pour l'approche JMS de la définition de ces options, voir [«Influence du rééquilibrage des applications dans IBM MQ classes for JMS»](#), à la page 457 pour plus d'informations.

D'autres environnements client ne prennent pas actuellement en charge la fourniture de cette structure lors de la connexion.

 *Configuration du comportement d'équilibrage à l'aide de l'interface MQI*

Pour influencer précisément le moment où IBM MQ rééquilibre les applications, certains environnements d'application client peuvent fournir des informations au moment de la connexion sur le modèle de messagerie utilisé.

Dans l'interface MQI, la structure des options d'équilibrage est appelée `MQBNO`.

Si aucune *option d'équilibrage* n'est fournie dans votre programme, les clients de support dérivent ces informations dans la [strophe d'application](#) ou dans la [stropheApplicationDefaults](#) du fichier `client.ini` déployé en même temps que l'application client.

Remarque : Ces strophes sont identiques, sauf que la version Application contient une zone **Name** pour identifier l'application à laquelle ces options s'appliquent.

Si l'une des formes de strophe est fournie, toutes les zones doivent être présentes, sauf **BalanceOptions** qui est supposé être none s'il n'est pas explicitement défini.

L'ordre de préférence pour la fourniture des options est le suivant:

1. Une structure MQBNO est fournie par l'application sur CONNX et utilisée dans son intégralité
2. Ou bien, la section Application correspondante, si elle est présente, est uniquement utilisée pour en générer une
3. Ou bien, la strophe ApplicationDefaults , si elle est présente, est utilisée uniquement pour en générer une
4. Ou bien, aucun flux MQBNO pour cette connexion.

Vous pouvez fournir trois éléments d'information clés à partir de la structure MQBNO ou du fichier client.ini :

1. **ApplicationType** ou modèle d'application.

Cette zone indique à IBM MQ le modèle général de l'activité IBM MQ à laquelle participe cette application.

Trois types d'application sont pris en charge:

Simple

Aucune règle spécifique ne doit être appliquée au-delà des valeurs par défaut décrites dans [«Comportement d'équilibrage d'application par défaut»](#), à la page 453 .

Demande - Réponse

Après chaque appel MQPUT, un appel MQGET correspondant est attendu pour un message de réponse. Voir [«Équilibrage demande-réponse»](#), à la page 453 pour plus de détails.

Client géré

Les demandes de rééquilibrage sont toujours envoyées immédiatement au client, qui les rééquilibre à un point qu'il juge approprié, par exemple, l'adaptateur de ressources JEE s'enregistrerait de cette manière.

2. Le **Timeout** après lequel le rééquilibrage peut interrompre l'activité de l'application
3. Spécifique **BalanceOptions**

Exemples de cas où votre application peut être rééquilibrée

Exemple 1

Vous avez écrit une application qui place les messages sous un point de synchronisation et valide le lot de messages en émettant un appel MQCMIT. Une fois l'appel MQCMIT terminé, l'application commence à placer les messages sous un nouveau point de synchronisation.

Configuration IBM MQ suggérée

Options par défaut suffisantes

Résultat

Une instance d'application est déplacée après la réussite (ou l'échec) d'un appel MQCMIT, une fois que le nombre de transactions configuré a été atteint.

Par défaut, si un lot de messages dépasse 10 secondes, il peut être annulé si un rééquilibrage a été demandé. Si vous prévoyez que les transactions dépassent régulièrement cette limite et que vous l'autorisez, vous pouvez étendre le **Timeout** de manière appropriée.

Exemple 2

Vous avez écrit une application qui insère un message dans une instance de file d'attente de cluster et une autre application répond à une file d'attente dynamique temporaire locale avec un message, après avoir traité la demande. Lorsque la demande a été lue de façon destructive à partir de la file d'attente locale, l'application insère son message de demande suivant.

Configuration IBM MQ suggérée

Définissez [Type](#) sur MQBNO_BALTYPE_REQREP

Résultat

Une instance d'application est déplacée lorsqu'une application effectue un appel MQGET, auquel cas l'instance d'application est déplacée vers un autre gestionnaire de files d'attente. Tous les appels MQPUT suivants sont exécutés sur le nouveau gestionnaire de files d'attente.

MQBNO

ApplicationType

Multi *Influence du rééquilibrage des applications dans .NET*

Depuis la IBM MQ 9.3.0, des constantes supplémentaires sont disponibles pour vous permettre de définir les propriétés de l'option d'équilibrage à l'aide d'une table de hachage de l'application lorsque vous utilisez la classe `MQQueueManager` pour vous connecter au gestionnaire de files d'attente.

Les constantes suivantes sont celles que vous utilisez pour influencer l'équilibrage des applications dans .NET:

Rééquilibrage du type d'application

Type d'action d'équilibrage ; représenté par la constante

MQC.BALANCING_APPLICATION_TYPE_PROPERTY

- Vous devez utiliser cette propriété pour définir la zone **ApplicationType** de la structure MQBNO.

Vous devez définir des valeurs de type entier et les valeurs possibles sont les suivantes:

MQC.BALANCING_APPLICATION_TYPE_SIMPLE

Équilibrage simple ; aucune règle spécifique n'est appliquée en plus de celles décrites dans «[Influence du rééquilibrage des applications dans les clusters uniformes](#)», à la page 452. Il s'agit de la valeur par défaut.

MQC.BALANCING_APPLICATION_TYPE_REQUEST_REPLY

Équilibrage de demande-réponse ; après chaque appel MQPUT, un appel MQGET correspondant est attendu pour un message de réponse. L'équilibrage est différé jusqu'à ce qu'un tel message soit reçu ou que le message de demande EXPIRY ait été dépassé.

Si la reconnexion est activée par l'application et que cette propriété n'est pas définie,

MQC.BALANCING_APPLICATION_TYPE_SIMPLE est utilisé

Options de rééquilibrage

Options d'équilibrage définies par l'application émettrice ; représentées par la constante

MQC.BALANCING_OPTIONS_PROPERTY

- Vous devez utiliser cette propriété pour définir la zone **BalanceOptions** de la structure MQBNO.

Vous devez définir des valeurs de type entier et les valeurs possibles sont les suivantes:

MQC.BALANCING_OPTIONS_NONE

Aucune option n'est définie. Il s'agit de la valeur par défaut

MQC.BALANCING_OPTIONS_IGNORE_TRANSACTIONS

La définition de cette option permet de rééquilibrer les applications même si elles se trouvent au milieu d'une transaction.

Si la reconnexion est activée par l'application et que cette propriété n'est pas définie,

MQC.BALANCING_OPTIONS_NONE est utilisé.

Rééquilibrage du délai d'attente

Délai après lequel le rééquilibrage peut interrompre l'activité de l'application ; représenté par la constante **MQC.BALANCING_TIMEOUT_PROPERTY**

- Vous devez utiliser cette propriété pour définir la zone Délai d'attente de la structure MQBNO.

Vous devez définir des valeurs de type entier et les valeurs possibles sont les suivantes:

MQC.BALANCING_TIMEOUT_AS_DEFAULT

Valeur de délai d'attente par défaut définie. Il s'agit de la valeur par défaut

MQC.BALANCING_TIMEOUT_IMMEDIATE

Dépassement immédiat du délai d'attente

MQC.BALANCING_TIMEOUT_NEVER

Aucun délai d'attente n'est dépassé

Remarque : Vous devez indiquer une seule valeur parmi les valeurs définies ou une valeur comprise entre 0 et 999999999 secondes.

Déploiement d'applications client flexibles et évolutives

MQBNO

V 9.4.0 **Multi** *Influence du rééquilibrage des applications dans IBM MQ classes for JMS*

Depuis la IBM MQ 9.4.0, des constantes supplémentaires sont disponibles pour vous permettre de définir les propriétés d'option d'équilibrage sur un **ConnectionFactory**. Ces constantes ne sont applicables que si **WMQ_PROVIDER_VERSION** est défini sur 7. Les applications Request_reply dans un cluster uniforme doivent permettre la possibilité de réponses manquées.

- «Les constantes disponibles», à la page 457.
- «Risque de perte de messages lors de l'équilibrage des applications REQUEST_REPLY», à la page 458.

Les constantes disponibles

Les constantes suivantes sont celles que vous utilisez pour influencer l'équilibrage des applications dans IBM MQ classes for JMS:

Rééquilibrage du type d'application

Type d'action d'équilibrage ; représenté par la constante **WMQConstants.WMQ_BALANCING_APPLICATION_TYPE**

- Vous devez utiliser cette propriété pour définir la zone **ApplicationType** de la structure MQBNO .

Vous devez définir des valeurs de type entier. Les valeurs possibles sont les suivantes:

WMQConstants.WMQ_BALANCING_APPLICATION_TYPE_SIMPLE (par défaut)

Équilibrage simple ; aucune règle spécifique n'est appliquée en plus de celles décrites dans «Influence du rééquilibrage des applications dans les clusters uniformes», à la page 452.

WMQConstants.WMQ_BALANCING_APPLICATION_TYPE_REQUEST_REPLY

Équilibrage de demande-réponse ; après chaque appel **MQPUT** , un appel **MQGET** correspondant est attendu pour un message de réponse. L'équilibrage est différé jusqu'à ce qu'un tel message soit reçu ou que le message de demande **EXPIRY** ait été dépassé.

Si la reconnexion est activée par l'application et que cette propriété n'est pas définie, **WMQConstants.WMQ_BALANCING_APPLICATION_TYPE_SIMPLE** est utilisé.

Options de rééquilibrage

Options d'équilibrage définies par l'application émettrice ; représentées par la constante **WMQConstants.WMQ_BALANCING_OPTIONS**

- Vous devez utiliser cette propriété pour définir la zone **BalanceOptions** de la structure MQBNO .

Vous devez définir des valeurs de type entier. Les valeurs possibles sont les suivantes:

WMQConstants.WMQ_BALANCING_OPTIONS_NONE (par défaut)

Aucune option n'est définie.

WMQConstants.WMQ_BALANCING_OPTIONS_IGNORE_TRANSACTIONS

La définition de cette option permet de rééquilibrer les applications même en milieu de transaction.

Si la reconnexion est activée par l'application et que cette propriété n'est pas définie, **WMQConstants.WMQ_BALANCING_OPTIONS_NONE** est utilisé.

Rééquilibrage du délai d'attente

Délai après lequel le rééquilibrage peut interrompre l'activité de l'application ; représenté par la constante **WMQConstants.WMQ_BALANCING_TIMEOUT**

- Vous devez utiliser cette propriété pour définir la zone **Timeout** de la structure MQBNO .

Vous devez définir des valeurs de type entier. Les valeurs possibles sont les suivantes:

WMQConstants.WMQ_BALANCING_TIMEOUT_AS_DEFAULT (par défaut)

Valeur de délai d'attente par défaut définie. Par défaut, cette valeur est de 10 secondes.

WMQConstants.WMQ_BALANCING_TIMEOUT_IMMEDIATE

Le délai d'attente immédiat est dépassé.

WMQConstants.WMQ_BALANCING_TIMEOUT_NEVER

Aucun délai d'attente n'est dépassé.

une valeur comprise entre 1 et 999999999

Représente une valeur en secondes.

Remarque : Vous devez indiquer une seule valeur parmi les valeurs définies ou une valeur comprise entre 0 et 999999999 secondes.

Ces propriétés peuvent également être définies dans les représentations JNDI des fabriques de connexions à l'aide des interfaces JMSAdmin ou IBM MQ Explorer .

Risque de perte de messages lors de l'équilibrage des applications REQUEST_REPLY

Dans IBM MQ classes for JMS (et IBM MQ classes for Jakarta Messaging), la fonctionnalité de demande / réponse est implémentée en définissant la propriété **JMSReplyTo** sur le message de demande, qui est utilisé par l'application répondante pour déterminer si la réponse est envoyée. Dans les termes JMS , la propriété **JMSReplyTo** est un **Destination**.

Lorsqu'elle est convertie en opérations IBM MQ , la propriété **JMSReplyTo** est envoyée en tant qu'URI de file d'attente qualifié complet, identifiant une file d'attente sur un gestionnaire de files d'attente spécifique.

En raison de la nature asynchrone du traitement des reconnections d'équilibrage, une reconnection peut être lancée une fois que la propriété **JMSReplyTo** a été convertie en URI qualifié complet, mais avant que le message de demande ait été placé dans la file d'attente des demandes. Dans ces circonstances, l'application répondante peut envoyer sa réponse à la file d'attente de réponses d'origine sur le gestionnaire de files d'attente d'origine, mais l'application demandeuse peut maintenant attendre une réponse sur le nouveau gestionnaire de files d'attente.

Les applications Request_reply dans un cluster uniforme doivent donc permettre la possibilité de réponses manquées.

[Déploiement d'applications client flexibles et évolutives](#)

[MQBNO-Options d'équilibrage](#)

Multi *Limitations et remarques relatives aux clusters uniformes*

Limitations et autres points à prendre en compte lors de la configuration de clusters uniformes.

Remarque : Pour connaître les exigences générales lorsque vous configurez des clusters uniformes, voir aussi [«Création d'un nouveau cluster uniforme»](#), à la page 461.

Importance de l'uniformité entre les gestionnaires de files d'attente

Par défaut, toute application qui se déclare reconnectable peut être rééquilibrée dans un autre gestionnaire de files d'attente dans un cluster uniforme à tout moment. Cela signifie que toute ressource, par exemple, une file d'attente, une rubrique ou un enregistrement de droits d'accès requis par ces applications doit être déclarée sur tous les gestionnaires de files d'attente du cluster uniforme.

La cohérence de la configuration du gestionnaire de files d'attente n'est pas contrôlée. Il appartient à votre administrateur système de configurer les membres du cluster de sorte qu'ils aient une configuration similaire.

Toutefois, vous pouvez faciliter la cohérence en utilisant la fonction [Configuration automatique à partir d'un script MQSC au démarrage](#) pour partager des scripts MQSC qui définissent des objets pour le cluster et donc s'assurer que tous ont les mêmes définitions. Pour plus d'informations, voir [«Création d'un nouveau cluster uniforme»](#), à la page 461.

Cette uniformité s'étend aux gestionnaires de files d'attente de référentiel complet pour le cluster. Bien que pour les clusters IBM MQ traditionnels, il est souvent recommandé de séparer les référentiels complets sur des systèmes autonomes, dans un cluster uniforme, le modèle est que les référentiels complets participent pleinement aux charges de travail de cluster et d'application de processus aux côtés d'autres nœuds.

Chevauchement de clusters uniformes et de clusters IBM MQ traditionnels

Un gestionnaire de files d'attente de cluster uniforme peut participer à au plus un cluster uniforme, et il peut également être membre de n'importe quel nombre de clusters IBM MQ standard. Il peut être utile de penser que le cluster uniforme agit comme un gestionnaire de files d'attente unique dans le cluster plus large.

Notez les points suivants :

- Un gestionnaire de files d'attente de cluster uniforme faisant office de référentiel complet ne doit être qu'un référentiel complet pour le cluster uniforme lui-même.
- De même, les gestionnaires de files d'attente de référentiel partiel qui sont membres d'un cluster uniforme, mais qui peuvent également appartenir à un cluster IBM MQ traditionnel plus large, ne peuvent pas être utilisés en tant que référentiel en dehors du cluster uniforme.

Pour plus d'informations, voir [Comment choisir les gestionnaires de files d'attente de cluster devant contenir des référentiels complets](#).

La raison en est que les gestionnaires de files d'attente qui sont des référentiels complets pour une combinaison de clusters IBM MQ traditionnels et de clusters uniformes, encouragent une divergence des données contenues dans le cache de cluster entre les membres du cluster uniforme et, par conséquent, s'opposent à l'utilisation de la fonction de cluster uniforme comme prévu.

Pour remplacer un gestionnaire de files d'attente de référentiel complet unique par un cluster uniforme, séparez le référentiel complet du travail d'application en cours sur ce dernier et déplacez uniquement le travail d'application dans le cluster uniforme.

Lorsque vous utilisez des définitions automatiques pour des clusters uniformes, les canaux de cluster ne peuvent pas être partagés pour une utilisation dans d'autres clusters, c'est-à-dire que vous définissez l'attribut **CLUSTER** sur le cluster automatique et que l'attribut **CLUSNL** doit être vide.

Considérations relatives à l'équilibrage des applications

Les instances d'application ne sont pas toujours équilibrées de manière uniforme, en particulier dans les cas suivants:

- Lorsqu'il y a moins d'instances d'application que de gestionnaires de files d'attente dans le cluster.
- Pendant une courte période après la connexion des applications client au cluster ou leur sortie du cluster.

Pour éviter que les applications client ne soient trop souvent rééquilibrées, en particulier lorsque des connexions d'application sont établies, des limites sont définies sur la fréquence à laquelle le cluster uniforme demande que les applications client soient rééquilibrées. Après une période de forte activité de connexion ou de déconnexion, l'équilibrage des instances d'application restantes dans le cluster uniforme peut prendre plusieurs minutes.

Pour plus d'informations, voir [Traitement des incidents liés à l'équilibrage des applications](#).

Affinités d'application

Toutes les applications ne sont pas adaptées au rééquilibrage automatique dans un cluster uniforme. Seules les applications qui spécifient **MQCNO_RECONNECT** sont rééquilibrées. Les applications qui ont une affinité avec un gestionnaire de files d'attente particulier doivent spécifier l'option **MQCNO_NO_RECONNECT** ou **MQCNO_RECONNECT_Q_MGR**. Ce dernier permet la reprise en ligne à haute disponibilité mais pas le rééquilibrage.

Exemples d'applications qui créent une affinité implicite avec un gestionnaire de files d'attente:

- Applications qui créent des abonnements durables.
- Applications qui créent des files d'attente dynamiques permanentes, par exemple, pour recevoir des messages de réponse.
- Les applications qui attendent un ordre strict des messages ou qui requièrent tous les messages dans une séquence doivent être traitées par la même instance d'application ou les deux.

Ces applications doivent spécifier des options `MQCNO_NO_RECONNECT` ou `MQCNO_RECONNECT_Q_MGR` plutôt que `MQCNO_RECONNECT`.

Pour plus d'informations, voir [Options de reconnexion](#).

Disponibilité des messages

Alors que l'équilibrage des applications peut rééquilibrer les connexions en cas d'échec ou d'indisponibilité temporaire des gestionnaires de files d'attente, les clusters uniformes ne répliquent pas les données de message sur leurs membres. Pour la disponibilité des données, en cas de défaillance d'un noeud, chaque membre du cluster uniforme doit également être configuré pour être hautement disponible. De nombreuses solutions de réplication de données et de haute disponibilité sont disponibles et peuvent être combinées avec des clusters uniformes pour une disponibilité maximale des données et des services, par exemple:

- Stockage répliqué qui prend en charge une instance de conteneur qui est automatiquement redémarrée par l'orchestration de conteneur. Pour plus d'informations, voir [Gestionnaire de files d'attente résilient unique](#).
- Gestionnaires de files d'attente de données répliquées. Pour plus d'informations, voir [Haute disponibilité des gestionnaires de files d'attente de données répliquées](#).
- Gestionnaires de files d'attente multi-instance. Pour plus d'informations, voir [«Gestionnaires de files d'attente multi-instance»](#), à la page 547.
- Haute disponibilité native. Pour plus d'informations, voir [Native HA](#).
- IBM MQ Appliance Haute disponibilité. Pour plus d'informations, voir [Haute disponibilité](#).

Evolutivité et performances des clusters uniformes

Pour permettre une intégration et un partage plus étroits de l'état des applications entre les gestionnaires de files d'attente dans un cluster uniforme, un niveau d'intercommunication plus élevé est nécessaire que dans un cluster IBM MQ traditionnel. Par conséquent, la mise à l'échelle à un grand nombre de gestionnaires de files d'attente dans un cluster uniforme unique n'est pas recommandée car la communication supplémentaire a un impact négatif sur les performances.

Pour des raisons de performances et de gestion, il est préférable de considérer un cluster uniforme comme un gestionnaire de files d'attente classique unique qui fournit des messages à un certain nombre d'applications associées, mais qui n'est pas le seul service de messagerie d'une entreprise. Dans ce modèle, de petits nombres, jusqu'à 10, les gestionnaires de files d'attente sont généralement suffisants pour prendre en charge un grand nombre de connexions d'application client. L'équilibrage des applications simplifie le démarrage avec de petits nombres, par exemple 3 gestionnaires de files d'attente, et la mise à l'échelle en ajoutant des gestionnaires de files d'attente supplémentaires.



Avertissement : L'activation d'un comportement de cluster uniforme dans un cluster qui ne possède pas les caractéristiques recommandées, en particulier l'utilisation de clusters avec un grand nombre de gestionnaires de files d'attente, est susceptible d'avoir un impact grave sur les performances.

Concepts associés

[«Équilibrage automatique des applications»](#), à la page 447

L'équilibrage automatique des applications améliore considérablement la distribution et la disponibilité des applications en permettant à un cluster uniforme IBM MQ de gérer étroitement la distribution des

applications dans le cluster, plutôt que de s'appuyer sur la randomisation ou sur une épinglage manuel des applications à des gestionnaires de files d'attente spécifiques.

Création d'un cluster uniforme

Vous pouvez simplifier la création initiale d'un cluster uniforme, puis conserver la configuration identique entre les membres du cluster uniforme, à l'aide de la configuration automatique et de la prise en charge de la mise en cluster automatique.

Avant de commencer

Avant de créer un cluster uniforme, lisez la rubrique [«Limitations et remarques relatives aux clusters uniformes»](#), à la page 458.

Pourquoi et quand exécuter cette tâche

Vous indiquez qu'un cluster IBM MQ particulier doit être traité comme un cluster uniforme en fournissant dans le fichier `qm.ini` une section pour AutoCluster avec au moins **Type=Uniform** et **ClusterName=< nom de cluster uniforme >**.

Vous pouvez éventuellement configurer le cluster IBM MQ sous-jacent via la même section `.ini` à l'aide de la *création automatique de cluster*. Lorsque vous utilisez cette prise en charge de cluster automatique pour configurer le cluster, vous fournissez un fichier de configuration qui décrit le cluster et ses référentiels complets.

Si le gestionnaire de files d'attente en cours de démarrage est répertorié comme l'un des référentiels complets, il devient automatiquement un référentiel complet. De même, lorsque le canal récepteur de cluster est défini, les canaux émetteurs de cluster vers le ou les référentiels complets sont automatiquement définis.

Procédure

Pour exploiter une fonction supplémentaire qui nécessite un cluster uniforme, vous devez effectuer l'une des étapes suivantes:

- [Convertir un cluster existant en un cluster uniforme](#), qui répond au modèle décrit dans [«A propos des clusters uniformes»](#), à la page 445.
- [Créer un cluster uniforme](#) à cette fin.

Création d'un nouveau cluster uniforme

Comment créer un nouveau cluster uniforme.

Procédure

1. Créez un fichier qui décrit comment vous souhaitez que le cluster lui-même se présente en termes de référentiels complets.

Comme pour tout cluster, deux référentiels complets agissent comme des magasins centraux d'informations sur le cluster.

Plus précisément, vous devez décrire les noms et les noms de connexion des deux référentiels complets de ce cluster.

Remarque : Cette opération est effectuée avant la création de tout élément (y compris les gestionnaires de files d'attente) et le processus suivant inclut la création de ces gestionnaires de files d'attente.

Par exemple, imaginez que vous configurez un cluster uniforme appelé UNICLUS, avec les membres de gestionnaire de files d'attente QMA, QMB, QMC et QMD. Dans cet exemple, QMA et QMB seront

les référentiels complets, avec QMC et QMD comme référentiels partiels. Exemple de fichier de configuration, uniclus.ini:

```
AutoCluster:
  Repository2Conname=QMA.dnsname(1414)
  Repository2Name=QMA
  Repository1Conname=QMB.dnsname(1414)
  Repository1Name=QMB
  ClusterName=UNICLUS
  Type=Uniform
```

Les zones **RepositoryNConname** sont utilisées comme attribut *conname* pour les autres membres de cluster afin de leur définir des émetteurs de cluster (CLUSDR). Elles peuvent être une liste de connexion pour un gestionnaire de files d'attente multi-instance et inclure éventuellement le port.

2. Créez un exemple de fichier de configuration, uniclus.mqsc, qui contient les définitions MQSC que vous souhaitez appliquer à tous les membres du cluster.

Une ligne obligatoire est requise dans ce fichier, qui est une définition d'un canal récepteur de cluster (CLUSRCVR), avec un attribut CLUSTER du nom de cluster automatique (généralement via l'insertion + AUTOCL +) et un nom de canal qui inclut l'insertion + QMNAME +.

Il décrit comment les autres membres du cluster uniforme se connectent à chaque gestionnaire de files d'attente et est utilisé comme modèle de connexion aux autres gestionnaires de files d'attente. Voici un exemple de définition:

```
define channel('+AUTOCL+_QMNAME+') chltype(clusrcvr) trdtype(tcp)
conname(+CONNAME+) cluster('+AUTOCL+') replace
```

Lorsque des clusters automatiques sont configurés, la définition d'un canal récepteur de cluster peut utiliser des insertions supplémentaires dans les zones CLUSTER, CONNAME et CHANNEL pour permettre à la définition d'être identique sur tous les gestionnaires de files d'attente du cluster uniforme. Seront perdus :

+ AUTOCL +

Nom de cluster automatique

+ QMNAME +

Nom du gestionnaire de files d'attente en cours de création

+ CONNAME +

Variable définie lors de la création du gestionnaire de files d'attente, à l'aide du paramètre **-iv** ou dans la section Variables qm.ini, à utiliser dans la chaîne de paramètres du nom de connexion. Le nom de la variable peut être n'importe quelle valeur.

N'oubliez pas que les noms de canal sont limités à 20 caractères, de sorte que la valeur avec les insertions dans, ainsi que lorsque les insertions sont remplacées, doit tenir compte de cette limitation. Un exemple de fichier peut se présenter comme suit:

```
*#####
* Compulsory section for all uniform cluster queue managers
*#####
define channel('+AUTOCL+_QMNAME+') chltype(clusrcvr) trdtype(tcp) conname(+CONNAME+)
cluster('+AUTOCL+') replace
*
*#####
* Configuration for all queue managers
*#####
define QL(APPQ) maxdepth(9999999) replace
define QL(APPQ2) maxdepth(9999999) replace
define channel(CLIENTCHL) chltype(svrconn) trdtype(tcp) replace
```

3. Mettez ces deux fichiers à disposition sur chaque machine qui hébergera un membre de cluster uniforme.

Par exemple, /shared/uniclus.ini et /shared/uniclus.mqsc.

4. Sur chacune de ces machines, créez le gestionnaire de files d'attente.

Sur la ligne de commande, entrez:

- Demande de démarrage d'un programme d'écoute sur le port attendu
- Une demande de configuration INI automatique (**-ii**) pointant vers le fichier de configuration de cluster automatique (uniclus.ini)
- Une demande de configuration MQSC automatique (**-ic**) pointant vers le fichier de configuration MQSC qui inclut une définition CLUSRCVR pour le cluster uniforme.
- CONNNAME pour ce gestionnaire de files d'attente.

Sur l'hôte pour QMA:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv  
CONNNAME=QMA.dnsname(1414) QMA  
strmqm QMA
```

Chaque gestionnaire de files d'attente du cluster uniforme est créé avec une ligne de commande presque identique-toutes les différences entre le référentiel complet et le référentiel partiel sont gérées automatiquement pour un cluster uniforme.

Sur l'hôte pour QMB:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv  
CONNNAME=QMB.dnsname(1414) QMB  
strmqm QMB
```

Sur l'hôte pour QMC:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv  
CONNNAME=QMC.dnsname(1414) QMC  
strmqm QMC
```

Sur l'hôte pour QMD:

```
crtmqm -p 1414 -ii /shared/uniclus.ini -ic /shared/uniclus.mqsc -iv  
CONNNAME=QMD.dnsname(1414) QMD  
strmqm QMD
```

Ce qui se passe automatiquement:

Lorsque le gestionnaire de files d'attente est démarré, les définitions du fichier `uniclus.ini` sont appliquées au fichier `qm.ini`. Pour plus d'informations, voir [«Configuration automatique de qm.ini au démarrage»](#), à la page 112. La définition **AutoCluster** est ajoutée dans le fichier `qm.ini`.

Si le gestionnaire de files d'attente est nommé dans la section **AutoCluster** comme l'un des référentiels complets, il est automatiquement converti en référentiel complet, comme pour la commande MQSC ALTER QMGR REPOS (**ClusterName**), sinon il est converti en référentiel partiel, comme pour la commande MQSC ALTER QMGR REPOS (").

Lorsque la définition du canal récepteur de cluster pour le cluster automatique est traitée, les canaux émetteurs de cluster sont définis à partir de ce gestionnaire de files d'attente vers tous les référentiels complets dans la section **AutoCluster** (à l'exclusion du gestionnaire de files d'attente local s'il s'agit de l'un des référentiels complets). Ces canaux émetteurs héritent de tous les attributs de canal communs du récepteur de cluster local qui a été défini.



Avertissement : Bien que les canaux soient créés sans intervention manuelle, il s'agit d'objets de canal d'administration qui peuvent être affichés et gérés comme pour toute autre définition de canal. Vous ne devez pas confondre ces objets avec des canaux émetteurs de cluster 'définis automatiquement', créés de manière transitoire et à la demande par le cluster pour acheminer le trafic des messages.

Que faire ensuite

Vérification de la configuration de cluster uniforme

Lorsque le paramètre **ClusterName** est correctement défini et que le gestionnaire de files d'attente est membre du cluster nommé, le message AMQ9883 est émis pour confirmer que le cluster est désormais identifié comme un cluster uniforme.

Vous pouvez ensuite utiliser des fonctions de cluster uniformes, telles que l'équilibrage automatique des applications. Lors du démarrage du gestionnaire de files d'attente, si ce paramètre a été défini, mais que le nom n'est pas un nom de cluster IBM MQ valide, il est ignoré et le message d'erreur AMQ9882 est émis.

Si le nom est un nom de cluster valide, mais qu'aucun canal de cluster n'existe pour le cluster identifié, le message d'avertissement AMQ9881 est émis dans le journal des erreurs du gestionnaire de files d'attente pour permettre à votre administrateur d'identifier et de corriger cette situation.

Vérification de la configuration de cluster automatisée

Si vous avez utilisé le support de cluster automatique pour configurer le cluster uniforme, vous pouvez vérifier que les gestionnaires de files d'attente spécifiés en tant que référentiels complets sont désormais correctement configurés en tant que tels, à l'aide des commandes runmqsc:

```
QMA:
  1 : dis qmgr repos
AMQ8408I: Display Queue Manager details.
      QMNAME(QMA)                REPOS(UNICLUS)
```

Alors que les référentiels partiels ne sont pas configurés en tant que référentiels:

```
QMC:
  1 : dis qmgr repos
AMQ8408I: Display Queue Manager details.
      QMNAME(QMC)                REPOS( )
```


En outre, vous devriez être en mesure de voir que les canaux émetteurs de cluster (CLUSSDR) ont été configurés à partir de chaque gestionnaire de files d'attente vers les autres référentiels complets, à l'aide du nom de canal du fichier MQSC de configuration:

```
QMA:
  1 : dis chl(UNICLUS*) conname
AMQ8414I: Display Channel details.
      CHANNEL(UNICLUS_QMA)        CHLTYPE(CLUSRCVR)
      CONNAME(QMA.dnsname(1414))
AMQ8414I: Display Channel details.
      CHANNEL(UNICLUS_QMB)        CHLTYPE(CLUSSDR)
      CONNAME(QMB.dnsname(1414))

QMC:
  1 : dis chl(UNICLUS*) conname
AMQ8414I: Display Channel details.
      CHANNEL(UNICLUS_QMA)        CHLTYPE(CLUSSDR)
      CONNAME(QMA.dnsname(1414))
AMQ8414I: Display Channel details.
      CHANNEL(UNICLUS_QMB)        CHLTYPE(CLUSSDR)
      CONNAME(QMB.dnsname(1414))
AMQ8414I: Display Channel details.
      CHANNEL(UNICLUS_QMC)        CHLTYPE(CLUSRCVR)
      CONNAME(QMC.dnsname(1414))
```

Concepts associés

[«A propos des clusters uniformes», à la page 445](#)

L'objectif d'un déploiement en cluster uniforme est que les applications puissent être conçues pour l'échelle et la disponibilité et qu'elles puissent se connecter à n'importe quel gestionnaire de files d'attente au sein du cluster uniforme. Cela permet de supprimer toute dépendance sur un gestionnaire de files d'attente spécifique, ce qui améliore la disponibilité et l'équilibrage de la charge de travail du trafic de messagerie.  Les clusters uniformes ne sont pas disponibles sur IBM MQ for z/OS; les groupes de partage de files d'attente fournissent de nombreuses fonctionnalités d'un cluster uniforme.

[«Limitations et remarques relatives aux clusters uniformes», à la page 458](#)

Limitations et autres points à prendre en compte lors de la configuration de clusters uniformes.

Vous pouvez utiliser cette procédure pour convertir un cluster existant en cluster uniforme.

Pourquoi et quand exécuter cette tâche

Si vous convertissez un cluster existant en cluster uniforme, vous devez vous assurer que toutes les définitions requises pour prendre en charge l'équilibrage des applications entre les gestionnaires de files d'attente existent sur tous les membres du cluster.

Procédure

1. Activez l'abonnement de publication IBM MQ , y compris l'abonnement de publication distant (en cluster) sur tous les gestionnaires de files d'attente.

Il s'agit d'une condition préalable à la fonctionnalité de cluster uniforme. Vous devez donc vous assurer que les attributs PSMODE et PSCLUS du gestionnaire de files d'attente sont tous deux définis sur la valeur par défaut ENABLED.

2. Ajoutez une section **AutoCluster** dans le fichier `qm.ini` au nom du cluster IBM MQ , tel qu'il est utilisé dans les définitions d'objet MQSC, telles que les canaux de cluster.

Par exemple, si le nom du cluster est UNICLUS, ajoutez ou modifiez la strophe AutoCluster dans vos fichiers `qm.ini` comme suit:


```
AutoCluster:
  ClusterName=UNICLUS
  Type=Uniform
```

3. Redémarrez les gestionnaires de files d'attente pour appliquer le nouveau paramètre.
4. Envisagez d'utiliser la configuration automatique comme mécanisme pour vous assurer que tous les membres de cluster uniforme ont la même configuration appliquée au démarrage.

Pour plus d'informations, voir [Configuration automatique à partir d'un script MQSC au démarrage](#) .

Concepts associés

«A propos des clusters uniformes», à la page 445

L'objectif d'un déploiement en cluster uniforme est que les applications puissent être conçues pour l'échelle et la disponibilité et qu'elles puissent se connecter à n'importe quel gestionnaire de files d'attente au sein du cluster uniforme. Cela permet de supprimer toute dépendance sur un gestionnaire de files d'attente spécifique, ce qui améliore la disponibilité et l'équilibrage de la charge de travail du trafic de messagerie.  Les clusters uniformes ne sont pas disponibles sur IBM MQ for z/OS; les groupes de partage de files d'attente fournissent de nombreuses fonctionnalités d'un cluster uniforme.

«Limitations et remarques relatives aux clusters uniformes», à la page 458

Limitations et autres points à prendre en compte lors de la configuration de clusters uniformes.

Utilisation de la configuration de cluster automatique

Vous configurez IBM MQ pour activer la configuration automatique en modifiant les informations de configuration `qm.ini` .

Remarque : Vous ne pouvez utiliser la section AutoCluster que pour les clusters uniformes.

Sections à configurer

Vous pouvez modifier les sections suivantes:

AutoConfig

Défini dans le fichier `qm.ini` . Lorsque le gestionnaire de files d'attente démarre, il identifie les fichiers de configuration automatique à appliquer.

Vous devez utiliser ce mécanisme pour distribuer une configuration de cluster identique lorsque des clusters uniformes sont utilisés.

AutoCluster

Défini dans le fichier `qm.ini`. Utilisé lorsque le gestionnaire de files d'attente commence à identifier si le cluster est membre d'un cluster automatique et peut identifier les référentiels complets du cluster.

Variables

Défini dans le fichier `qm.ini`. Contient des variables de gestionnaire de files d'attente.

Attributs de la section AutoConfig

Les deux attributs suivants sont autorisés dans la section AutoConfig :

MQSCConfig=< *chemin*>

Le chemin d'accès est soit un chemin d'accès complet au fichier, soit un chemin d'accès à un répertoire dans lequel tous les fichiers `*.mqsc` sont appliqués au gestionnaire de files d'attente, à chaque démarrage du gestionnaire de files d'attente.

Pour plus d'informations, voir [Configuration automatique à partir d'un script MQSC au démarrage](#).

IniConfig=< *chemin*>

Il s'agit soit d'un chemin d'accès complet au fichier, soit d'un chemin d'accès à un répertoire dans lequel tous les fichiers `*.ini` sont appliqués au fichier `qm.ini`, à chaque démarrage du gestionnaire de files d'attente.

Pour plus d'informations, voir [«Configuration automatique de qm.ini au démarrage»](#), à la page 112.

Ces attributs sont fréquemment utilisés dans le cadre de la configuration de clusters uniformes. Pour plus d'informations, voir [«Création d'un nouveau cluster uniforme»](#), à la page 461.

Exemple de section:

```
AutoConfig:
MQSCConfig=C:\MQ_Configuration\uniclus.mqsc
IniConfig=C:\MQ_Configuration\uniclus.ini
```

Attributs de la section AutoCluster

Les attributs suivants sont obligatoires pour la section AutoCluster :

Type =*Uniforme*

Indique le type de cluster automatique et la seule option valide est *Uniforme*, qui représente un cluster uniforme.

ClusterName=< *Chaîne*>

Nom du cluster, c'est-à-dire le nom de cluster automatique.

La présence des attributs ci-dessus permet l'équilibrage des applications pour des clusters uniformes. Pour plus de détails, voir [«Équilibrage automatique des applications»](#), à la page 447.

En outre, la configuration simplifiée d'un cluster peut être effectuée si le cluster est décrit dans cette section. Pour plus d'informations, voir [«Création d'un nouveau cluster uniforme»](#), à la page 461. Lorsque cette option est utilisée, vous pouvez nommer deux gestionnaires de files d'attente et leur attribuer des noms de connexion pour les référentiels complets de ce cluster automatique.

Les attributs suivants sont facultatifs pour la section AutoCluster, mais vous devez les fournir par paires:

RepositoryName1 =< *String*>

Il s'agit du nom du gestionnaire de files d'attente pour le premier référentiel complet du cluster automatique. Il peut s'agir du nom de ce gestionnaire de files d'attente ou d'un autre nom.

Repository1Conname=< *chaîne de nom de connexion* >

Il s'agit de la valeur du nom de connexion (CONNNAME) pour la façon dont les membres du cluster automatique doivent se connecter à ce gestionnaire de files d'attente.

En outre, vous pouvez identifier un deuxième référentiel complet pour le cluster:

Repository2Name=< Chaîne>

Repository2Conname=< chaîne de nom de connexion >

Exemple de section:

```
AutoCluster:  
Repository2Conname=myFR1.hostname(1414)  
Repository2Name=QMFR1  
Repository1Conname= myFR2.hostname(1414)  
Repository1Name=QMFR2  
ClusterName=UNICLUS  
Type=Uniform
```

Attributs de la strophe Variables

Une paire `attribute=value` est valide dans la zone d'attribut. Elles peuvent être fournies à l'aide de l'option de ligne de commande **-iv** de la commande [crtmqm](#) lorsque vous créez un gestionnaire de files d'attente.

Vous pouvez utiliser les attributs répertoriés dans la section Variables lors de la configuration de cluster automatique de CONNAME et les zones MQSC de nom de canal d'un canal récepteur de cluster.

Interruption d'un gestionnaire de files d'attente d'un cluster uniforme

Lors du fonctionnement normal d'un cluster uniforme, les instances d'application client reconnectables peuvent être automatiquement rééquilibrées à tout moment à n'importe quel gestionnaire de files d'attente du cluster. Si vous souhaitez empêcher les applications de se connecter à un gestionnaire de files d'attente particulier pendant un certain temps, par exemple lors des opérations de maintenance ou de l'identification des problèmes, utilisez la commande `SUSPEND QMGR`.

Exécutez la commande `SUSPEND QMGR CLUSTER (nom de cluster uniforme)`

Outre les effets habituels de l'interruption d'un cluster IBM MQ, dans un cluster uniforme, la commande `SUSPEND` empêche également les applications reconnectables d'être rééquilibrées avec ce gestionnaire de files d'attente.

Les connexions existantes de ce type au gestionnaire de files d'attente sont immédiatement rééquilibrées avec les autres gestionnaires de files d'attente disponibles dans le cluster lorsque la commande est émise.

Remarques :

- Lorsque des gestionnaires de files d'attente sont interrompus à partir d'un cluster, `DIS APSTATUS` les affiche sous la forme ACTIVE (NO), à l'exception du gestionnaire de files d'attente local, qui affiche toujours ACTIVE (YES) pour sa propre entrée de statut.
- Si tous les gestionnaires de files d'attente du cluster uniforme sont interrompus, les applications restent connectées à un ou plusieurs des gestionnaires de files d'attente suspendus.

Pour éviter que de nouvelles connexions soient ajoutées au gestionnaire de files d'attente en cours de maintenance, vous devez arrêter le ou les canaux de connexion serveur utilisés par vos applications client, par exemple en exécutant la commande `runmqsc` suivante:

```
STOP CHANNEL(surconn channel name)
```

Cela peut ne pas être possible si, par exemple, ces canaux sont également utilisés pour connecter les applications d'administration requises lors de la fenêtre de maintenance. Pour cette raison, le gestionnaire de files d'attente suspendu recherche régulièrement les applications reconnectables connectées.

Si des applications reconnectables sont présentes, elles sont rééquilibrées vers d'autres gestionnaires de files d'attente disponibles dans le cluster. La maintenance peut désormais être effectuée sur le gestionnaire de files d'attente suspendu.

Remarque : Les applications qui ne sont pas considérées comme mobiles ne sont affectées ni par la commande initiale, ni par les réexamens ultérieurs, et restent connectées au gestionnaire de files d'attente suspendu ; voir [MOVCOUNT](#) pour plus de détails.

Pour reprendre un gestionnaire de files d'attente suspendu:

1. Si nécessaire, démarrez le canal de connexion serveur pour reprendre l'acceptation des nouvelles connexions d'application en exécutant la commande suivante:

```
START CHANNEL(svrconn channel name)
```

2. Exécutez la commande **runmqsc** suivante:

```
RESUME QMGR CLUSTER(uniform cluster name)
```

Le gestionnaire de files d'attente reprend la communication avec le reste du cluster uniforme et, si nécessaire pour rétablir l'équilibre, les instances d'application client reconnectables sont redirigées vers ce gestionnaire de files d'attente.

Configuration de la messagerie de type publication/abonnement

Vous pouvez démarrer, arrêter et afficher le statut de la publication / abonnement en file d'attente. Vous pouvez également ajouter et supprimer des flux et ajouter et supprimer des gestionnaires de files d'attente à partir d'une hiérarchie de courtiers.

Procédure

- Pour plus d'informations sur le contrôle de la publication / abonnement en file d'attente, voir les sous-rubriques suivantes:
 - «Définition des attributs de messages de publication/d'abonnement en file d'attente», à la page 468
 - «Démarrage de la publication/l'abonnement en file d'attente», à la page 470
 - «Arrêt de la publication/l'abonnement en file d'attente», à la page 470
 - «Ajout d'un flux», à la page 471
 - «Suppression d'un flux», à la page 472
 - «Ajout d'un point d'abonnement», à la page 472
 - «Combinaison d'espaces de sujet dans des réseaux de publication / abonnement», à la page 481

Définition des attributs de messages de publication/d'abonnement en file d'attente

Vous contrôlez le comportement de certains attributs de message de publication / abonnement à l'aide d'attributs de gestionnaire de files d'attente. Les autres attributs que vous contrôlez dans la section *Courtier* du fichier `qm.ini`.

Pourquoi et quand exécuter cette tâche

Vous pouvez définir les attributs de publication / abonnement suivants: pour plus de détails, voir [Paramètres du gestionnaire de files d'attente](#)

Tableau 28. Paramètres de configuration de publication / abonnement	
Description	Nom du paramètre MQSC
Nombre de relances du message de commande	PSRTYCNT
Suppression du message en entrée non transmissible	PSNPMSG

Tableau 28. Paramètres de configuration de publication / abonnement (suite)

Description	Nom du paramètre MQSC
Comportement suite au message de réponse de commande non transmissible	PSNPRES
Traitement des messages de commande sous le contrôle d'un point de synchronisation	PSSYNCPT

La section Broker permet de gérer les paramètres de configuration suivants:

- `PersistentPublishRetry=yes | force`

Si vous indiquez `Oui`, si une publication d'un message persistant via l'interface de publication / abonnement en file d'attente échoue et qu'aucune réponse négative n'a été demandée, l'opération de publication est relancée.

Si vous avez demandé un message de réponse négative, la réponse négative est envoyée et aucune nouvelle tentative n'est effectuée.

Si vous spécifiez `Force`, si une publication d'un message persistant via l'interface de publication / abonnement en file d'attente échoue, l'opération de publication est relancée jusqu'à ce qu'elle soit correctement traitée. Aucune réponse négative n'est envoyée.

- `NonPersistentPublishRetry= yes | force`

Si vous spécifiez `Oui`, si une publication d'un message non persistant via l'interface de publication / abonnement en file d'attente échoue et qu'aucune réponse négative n'a été demandée, l'opération de publication est relancée.

Si vous avez demandé un message de réponse négative, la réponse négative est envoyée et aucune nouvelle tentative n'est effectuée.

Si vous avez spécifié `Force`, si une publication d'un message non persistant via l'interface de publication / abonnement en file d'attente échoue, l'opération de publication est relancée jusqu'à ce qu'elle soit correctement traitée. Aucune réponse négative n'est envoyée.

Remarque : Si vous souhaitez activer cette fonctionnalité pour les messages non persistants, vous devez également définir la valeur `NonPersistentPublishRetry` et vous assurer que l'attribut de gestionnaire de files d'attente **PSSYNCPT** est défini sur `Yes`.

Cette opération peut également avoir un impact sur les performances du traitement des publications non persistantes car **MQGET** de la file d'attente `STREAM` se trouve désormais sous le point de synchronisation.

- `PublishBatchTaille =nombre`

Le courtier traite normalement les messages de publication dans le point de synchronisation. Il peut s'avérer inefficace de valider chaque publication individuellement et, dans certaines circonstances, le courtier peut traiter plusieurs messages de publication dans une seule unité de travail. Ce paramètre indique le nombre maximal de messages de publication pouvant être traités dans une seule unité d'oeuvre

La valeur par défaut pour `PublishBatchSize` est 5.

- `PublishBatchInterval =numéro`

Le courtier traite normalement les messages de publication dans le point de synchronisation. Il peut s'avérer inefficace de valider chaque publication individuellement et, dans certaines circonstances, le courtier peut traiter plusieurs messages de publication dans une seule unité de travail. Ce paramètre indique la durée maximale (en millisecondes) entre le premier message d'un lot et toute publication ultérieure incluse dans le même lot.

Un intervalle de traitement par lots de 0 indique que jusqu'à `PublishBatchSize` messages peuvent être traités, à condition que les messages soient disponibles immédiatement.

La valeur par défaut de `PublishBatchInterval` est zéro.

Procédure

Utilisez IBM MQ Explorer, des commandes programmables ou la commande **runmqsc** pour modifier les attributs du gestionnaire de files d'attente qui contrôlent le comportement de la publication / abonnement.

Exemple

```
ALTER QMGR PSNPRES(SAFE)
```

Démarrage de la publication/l'abonnement en file d'attente

Vous démarrez la publication / l'abonnement en file d'attente en définissant l'attribut `PSMODE` du gestionnaire de files d'attente.

Avant de commencer

Lisez la description de [PSMODE](#) pour comprendre les trois modes de publication / abonnement:

- COMPAT
- Désactivé
- Activé

Pourquoi et quand exécuter cette tâche

Définissez l'attribut `PSMODE` de `QMGR` pour démarrer l'interface de publication / abonnement en file d'attente (également appelée courtier) ou le moteur de publication / abonnement (également appelé publication / abonnement version 7) ou les deux. Pour démarrer la publication / l'abonnement en file d'attente, vous devez définir `PSMODE` sur `ENABLED`. La valeur par défaut est `ENABLED`.

Procédure

Utilisez IBM MQ Explorer ou la commande **runmqsc** pour activer l'interface de publication / abonnement en file d'attente si l'interface n'est pas déjà activée.

Exemple

```
ALTER QMGR PSMODE (ENABLED)
```

Que faire ensuite

IBM MQ traite les commandes de publication / abonnement en file d'attente et les appels MQI (Message Queue Interface) de publication / abonnement.

Arrêt de la publication/l'abonnement en file d'attente

Vous arrêtez la publication / l'abonnement en file d'attente en définissant l'attribut `PSMODE` du gestionnaire de files d'attente.

Avant de commencer

Lisez la description de [PSMODE](#) pour comprendre les trois modes de publication / abonnement:

- COMPAT
- DESACTIVE
- Activée

Pourquoi et quand exécuter cette tâche

Définissez l'attribut QMGR PSMODE pour arrêter l'interface de publication / abonnement en file d'attente (également appelée courtier) ou le moteur de publication / abonnement (également appelé publication / abonnement version 7) ou les deux. Pour arrêter la publication / l'abonnement en file d'attente, vous devez définir PSMODE sur COMPAT. Pour arrêter entièrement le moteur de publication / abonnement, définissez PSMODE sur DISABLED.

Procédure

Utilisez IBM MQ Explorer ou la commande **runmqsc** pour désactiver l'interface de publication / abonnement en file d'attente.

Exemple

```
ALTER QMGR PSMODE (COMPAT)
```

Ajout d'un flux

Vous pouvez ajouter des flux manuellement pour permettre l'isolement des données entre les applications ou pour permettre l'interopération avec des hiérarchies de publication / abonnement IBM MQ.

Avant de commencer

Familiarisez-vous avec le fonctionnement des flux de publication / abonnement. Voir [Flux et rubriques](#).

Pourquoi et quand exécuter cette tâche

Utilisez la commande PCF, **runmqsc** ou IBM MQ Explorer pour effectuer ces étapes.

Remarque : Vous pouvez effectuer les étapes 1 et 2 dans n'importe quel ordre. N'effectuez l'étape 3 qu'une fois les étapes 1 et 2 terminées.

Procédure

1. Définissez une file d'attente locale portant le même nom que le flux de la version antérieure de IBM MQ.
2. Définissez une rubrique locale portant le même nom que le flux sur la version précédente de IBM MQ.
3. Ajoutez le nom de la file d'attente à la liste de noms, SYSTEM.QPUBSUB.QUEUE.NAMELIST
4. Répétez cette opération pour tous les gestionnaires de files d'attente de la version ultérieure de IBM MQ qui se trouvent dans la hiérarchie de publication / abonnement.

Ajout de 'Sport'

Dans l'exemple de partage du flux 'Sport', les gestionnaires de files d'attente de la version antérieure et les gestionnaires de files d'attente IBM MQ de la version ultérieure fonctionnent dans la même hiérarchie de publication / abonnement. Les gestionnaires de files d'attente de la version antérieure partagent un flux appelé 'Sport'. L'exemple montre comment créer une file d'attente et une rubrique sur les gestionnaires de files d'attente d'une version plus récente appelée 'Sport', avec une chaîne de rubrique 'Sport' qui est partagée avec le flux de gestionnaires de files d'attente d'une version antérieure 'Sport'.

Une application de publication de gestionnaire de files d'attente de version ultérieure, publiée dans la rubrique 'Sport', avec la chaîne de rubrique 'Soccer/Results', crée la chaîne de rubrique résultante 'Sport/Soccer/Results'. Sur les gestionnaires de files d'attente de la version plus récente, les abonnés à la rubrique 'Sport', avec la chaîne de rubrique 'Soccer/Results', reçoivent la publication.

Sur les gestionnaires de files d'attente de version antérieure, les abonnés au flux 'Sport', avec la chaîne de rubrique 'Soccer/Results', reçoivent la publication.

```
runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
define qlocal('Sport')
  1 : define qlocal('Sport')
AMQ8006: IBM MQ queue created.
define topic('Sport') topicstr('Sport')
  2 : define topic('Sport') topicstr('Sport')
AMQ8690: IBM MQ topic created.
alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
  3 : alter namelist(SYSTEM.QPUBSUB.QUEUE.NAMELIST) NAMES('Sport', 'SYSTEM.BROKER.DEFAULT.STREAM',
'SYSTEM.BROKER.ADMIN.STREAM')
AMQ8551: IBM MQ namelist changed.
```

Remarque : Vous devez fournir les noms existants dans l'objet liste de noms, ainsi que les nouveaux noms que vous ajoutez à la commande **alter namelist**.

Que faire ensuite

Les informations sur le flux sont transmises à d'autres courtiers de la hiérarchie.

Vous devez configurer chaque gestionnaire de files d'attente IBM MQ dans la hiérarchie manuellement.

Suppression d'un flux

Vous pouvez supprimer un flux d'un gestionnaire de files d'attente IBM MQ.

Avant de commencer

Avant de supprimer un flux, vous devez vous assurer qu'il ne reste plus d'abonnements au flux et mettre au repos toutes les applications qui utilisent le flux. Si les publications continuent de circuler dans un flux supprimé, l'administration doit déployer beaucoup d'efforts pour restaurer le système à un état de fonctionnement correct.

Procédure

1. Recherchez tous les courtiers connectés qui hébergent ce flux.
2. Annulez tous les abonnements au flux sur tous les courtiers.
3. Supprimez la file d'attente (portant le même nom que le flux) de la liste de noms, SYSTEM.QPUBSUB.QUEUE.NAMELIST.
4. Supprimez ou purgez tous les messages de la file d'attente portant le même nom que le flux.
5. Supprimez la file d'attente portant le même nom que le flux.
6. Supprimez l'objet de rubrique associé.

Que faire ensuite

Répétez les étapes 3 à 5 sur tous les autres gestionnaires de files d'attente IBM MQ connectés hébergeant le flux.

Ajout d'un point d'abonnement

Comment étendre une application de publication / abonnement en file d'attente existante que vous avez migrée depuis une version antérieure d'IBM Integration Bus avec un nouveau point d'abonnement.

Avant de commencer

1. Vérifiez que le point d'abonnement n'est pas déjà défini dans SYSTEM.QPUBSUB.SUBPOINT.NAMELIST.

2. Vérifiez s'il existe un objet de rubrique ou une chaîne de rubrique portant le même nom que le point d'abonnement.


Pourquoi et quand exécuter cette tâche

IBM MQ, les applications n'utilisent pas de points d'abonnement, mais elles peuvent interagir avec les applications existantes, à l'aide du mécanisme de migration de point d'abonnement.

Important : Le mécanisme de migration de point d'abonnement a été supprimé de IBM MQ 8.0. Si vous devez migrer vos applications existantes, vous devez effectuer les procédures décrites dans la documentation de votre version du produit avant de migrer vers la version la plus récente.

Il n'est pas nécessaire d'ajouter des points d'abonnement pour utiliser des applications de publication / abonnement intégrées écrites pour les versions de IBM MQ.

Procédure

1. Ajoutez le nom du point d'abonnement à `SYSTEM.QPUBSUB.SUBPOINT.NAMELIST`.
 -  Sous z/OS, **NLTYPE** est NONE, la valeur par défaut.
 - Répétez l'étape sur chaque gestionnaire de files d'attente connecté dans la même topologie de publication / abonnement.
2. Ajoutez un objet de rubrique, de préférence en lui attribuant le nom du point d'abonnement, avec une chaîne de rubrique correspondant au nom du point d'abonnement.
 - Si le point d'abonnement se trouve dans un cluster, ajoutez l'objet de rubrique en tant que rubrique de cluster sur l'hôte de rubrique de cluster.
 - Si un objet de rubrique existe avec la même chaîne de rubrique que le nom du point d'abonnement, utilisez l'objet de rubrique existant. Vous devez comprendre les conséquences de la réutilisation d'une rubrique existante par le point d'abonnement. Si la rubrique existante fait partie d'une application existante, vous devez résoudre la collision entre deux rubriques portant le même nom.
 - S'il existe un objet de rubrique portant le même nom que le point d'abonnement, mais avec une chaîne de rubrique différente, créez une rubrique portant un nom différent.
3. Définissez l'attribut **Topic** WILDCARD sur la valeur BLOCK.

Le blocage des abonnements à # ou * isole les abonnements génériques aux points d'abonnement. Voir [Caractères génériques et points d'abonnement](#).
4. Définissez les attributs dont vous avez besoin dans l'objet de rubrique.

Exemple

L'exemple illustre un fichier de commandes **runmqsc** qui ajoute deux points d'abonnement, USD et GBP.

```
DEFINE TOPIC(USD) TOPICSTR(USD)
DEFINE TOPIC(GBP) TOPICSTR(GBP) WILDCARD(BLOCK)
ALTER NL(SYSTEM.QPUBSUB.SUBPOINT.NAMELIST) NAMES(SYSTEM.BROKER.DEFAULT.SUBPOINT, USD, GBP)
```

Remarque :

1. Incluez le point d'abonnement par défaut dans la liste des points d'abonnement ajoutés à l'aide de la commande **ALTER**. **ALTER** supprime les noms existants dans la liste de noms.
2. Définissez les rubriques avant de modifier la liste de noms. Le gestionnaire de files d'attente ne vérifie la liste de noms que lorsque le gestionnaire de files d'attente démarre et lorsque la liste de noms est modifiée.

Configuration des réseaux de publication/abonnement réparti

Les gestionnaires de files d'attente connectés ensemble dans une topologie de publication / abonnement distribuée partagent un espace de sujet fédéré commun. Les abonnements créés sur un gestionnaire

de files d'attente peuvent recevoir des messages publiés par une application connectée à un autre gestionnaire de files d'attente dans la topologie.

Vous pouvez contrôler l'étendue des espaces de sujet créés en connectant des gestionnaires de files d'attente dans des clusters ou des hiérarchies. Dans un cluster de publication / abonnement, un objet de rubrique doit être en cluster pour chaque branche de l'espace de sujet qui doit couvrir le cluster. Dans une hiérarchie, chaque gestionnaire de files d'attente doit être configuré pour identifier son parent dans la hiérarchie.

Vous pouvez contrôler davantage le flux des publications et des abonnements dans la topologie en choisissant si chaque publication et abonnement est local ou global. Les publications et les abonnements locaux ne sont pas propagés au-delà du gestionnaire de files d'attente auquel le diffuseur de publications ou l'abonné est connecté.

Concepts associés

[Réseaux de publication/abonnement répartis](#)

[Portée de la publication](#)

[Portée de l'abonnement](#)

[Espaces de sujet](#)

Tâches associées

[Définition de rubriques de cluster](#)

Configuration d'un cluster de publication / abonnement

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER**. Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLROUTE**.

Avant de commencer

Certaines configurations de cluster ne peuvent pas prendre en charge les surcharges de la publication / l'abonnement routé directement. Avant d'utiliser cette configuration, explorez les remarques et les options détaillées dans [Conception de clusters de publication / abonnement](#).

Pour que les modifications apportées à un cluster soient propagées dans le cluster, au moins un référentiel complet doit toujours être disponible. Assurez-vous que vos référentiels sont disponibles avant de commencer cette tâche.

Voir aussi [Routage pour les clusters de publication / abonnement: remarques sur le comportement](#).

Scénario :

- Le cluster INVENTORY a été configuré comme décrit dans «Ajout d'un gestionnaire de files d'attente à un cluster», à la page 343. Il contient trois gestionnaires de files d'attente: LONDON et NEWYORK contiennent tous deux des référentiels complets, PARIS un référentiel partiel.

Pourquoi et quand exécuter cette tâche

Lorsque vous définissez une rubrique sur un gestionnaire de files d'attente dans un cluster, vous devez spécifier si la rubrique est une rubrique de cluster et (si tel est le cas) le routage dans le cluster pour les publications et les abonnements pour cette rubrique. Pour que la rubrique devienne une rubrique de cluster, vous devez configurer la propriété **CLUSTER** sur l'objet TOPIC avec le nom du cluster. En définissant une rubrique de cluster sur un gestionnaire de files d'attente dans le cluster, vous rendez la rubrique disponible pour l'ensemble du cluster. Pour choisir le routage des messages à utiliser dans le cluster, définissez la propriété **CLROUTE** sur l'objet TOPIC sur l'une des valeurs suivantes:

- **DIRECT**
- **TOPICHOST**

Par défaut, le routage de rubrique est **DIRECT**. Lorsque vous configurez une rubrique de cluster routée directement sur un gestionnaire de files d'attente, tous les gestionnaires de files d'attente du cluster détectent les autres gestionnaires de files d'attente de ce cluster. Lorsqu'il effectue des opérations de

publication et d'abonnement, chaque gestionnaire de files d'attente peut se connecter directement à n'importe quel autre gestionnaire de files d'attente du cluster. Voir [Clusters de publication / abonnement routés directement](#).

A partir de IBM MQ 8.0, vous pouvez à la place configurer le routage de rubrique en tant que **TOPICHOST**. Lorsque vous utilisez le routage via un hôte de rubrique, tous les gestionnaires de files d'attente du cluster détectent les gestionnaires de files d'attente de cluster qui hébergent la définition de rubrique routée (à savoir les gestionnaires de files d'attente sur lesquels vous avez défini l'objet de rubrique). Lorsque vous effectuez des opérations de publication et d'abonnement, les gestionnaires de files d'attente du cluster ne se connectent qu'à ces gestionnaires de files d'attente hôte de rubrique et ne se connectent pas directement les uns aux autres. Les gestionnaires de files d'attente hôte de rubrique sont responsables du routage des publications depuis les gestionnaires de files d'attente sur lesquels les publications sont publiées vers les gestionnaires avec les abonnements correspondants. Voir [Clusters de publication / abonnement routés par l'hôte de rubrique](#).

Remarque : Une fois qu'un objet de rubrique a été mis en cluster (via la définition de la propriété **CLUSTER**), vous ne pouvez pas modifier la valeur de la propriété **CLROUTE**. La mise en cluster de l'objet doit être annulée (**CLUSTER** associé à ' ') pour que vous puissiez changer la valeur. L'annulation de la mise en cluster d'une rubrique convertit la définition de rubrique en rubrique locale, ce qui génère une période au cours de laquelle les publications ne sont pas distribuées aux abonnements dans les gestionnaires de files d'attente éloignées ; tenez compte de cette remarque lorsque vous procédez à cette modification. Voir la rubrique relative à [l'effet de la définition d'une rubrique autre qu'une rubrique de cluster avec le même nom qu'une rubrique de cluster à partir d'un autre gestionnaire de files d'attente..](#) Si vous tentez de modifier la valeur de la propriété **CLROUTE** alors qu'elle est en cluster, le système génère une exception MQRCCF_CLROUTE_NOT_ALTERABLE .

Procédure

1. Choisissez un gestionnaire de files d'attente pour héberger votre rubrique.

Tout gestionnaire de files d'attente de cluster peut héberger une rubrique. Choisissez l'un des trois gestionnaires de files d'attente (LONDON, NEWYORK ou PARIS) et configurez les propriétés de l'objet TOPIC . Si vous prévoyez d'utiliser le routage direct, le gestionnaire de files d'attente que vous choisissez n'est pas opérationnel. Si vous prévoyez d'utiliser le routage via l'hôte de rubrique, le gestionnaire de files d'attente choisi a des responsabilités supplémentaires pour le routage des publications. Par conséquent, pour le routage via un hôte de rubrique, choisissez un gestionnaire de files d'attente hébergé sur l'un de vos systèmes les plus puissants et disposant d'une bonne connectivité réseau.

2. [Définissez une rubrique sur un gestionnaire de files d'attente.](#)

Pour faire de la rubrique une rubrique de cluster, incluez le nom du cluster lorsque vous définissez la rubrique et définissez le routage à utiliser pour les publications et les abonnements de cette rubrique. Par exemple, pour créer une rubrique de cluster de routage direct sur le gestionnaire de files d'attente LONDON , créez la rubrique comme suit:

```
DEFINE TOPIC(INVENTORY) TOPICSTR('/INVENTORY') CLUSTER(INVENTORY) CLROUTE(DIRECT)
```

En définissant une rubrique de cluster sur un gestionnaire de files d'attente dans le cluster, vous rendez la rubrique disponible pour l'ensemble du cluster.

Pour plus d'informations sur l'utilisation de **CLROUTE**, voir [DEFINE TOPIC \(CLROUTE\)](#) et [Routing for publish / subscribe clusters: Notes on behavior](#).

Résultats

Le cluster est prêt à recevoir des publications et des abonnements pour la rubrique.

Que faire ensuite

Si vous avez configuré un cluster de publication / abonnement routé par un hôte de rubrique, vous souhaitez probablement ajouter un deuxième hôte de rubrique pour cette rubrique. Voir [«Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique»](#), à la page 478.

Si vous disposez de plusieurs clusters de publication / abonnement distincts, par exemple parce que votre organisation est géographiquement dispersée, vous pouvez propager certaines rubriques de cluster dans tous les clusters. Pour ce faire, vous pouvez connecter les clusters dans une hiérarchie. Voir [«Combinaison des espaces de sujet de plusieurs clusters»](#), à la page 483. Vous pouvez également contrôler les publications qui circulent d'un cluster à un autre. Voir [«Combinaison et isolement d'espaces de sujet dans plusieurs clusters»](#), à la page 485.

Concepts associés

[Combinaison des portées de publication et d'abonnement](#)

A partir de IBM WebSphere MQ 7.0, la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

[Combinaison d'espaces de sujet dans des réseaux de publication / abonnement](#)

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

Tâches associées

[Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente](#)

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

[Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique](#)

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

[Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement](#)

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

[Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement](#)

Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

[Conception des clusters de publication/abonnement](#)

[Identification et résolution des problèmes de publication / abonnement distribué](#)

[Blocage de la publication / l'abonnement en cluster](#)

Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

Pourquoi et quand exécuter cette tâche

Vous pouvez avoir plusieurs définitions du même objet de rubrique de cluster dans un cluster. Il s'agit d'un état normal pour un cluster routé par un hôte de rubrique et d'un état inhabituel pour un cluster routé directement. Pour plus d'informations, voir [Définitions multiples de rubrique de cluster de même nom](#).

Pour déplacer une définition de rubrique de cluster vers un gestionnaire de files d'attente différent dans le cluster sans interrompre le flux de publications, procédez comme suit. La procédure déplace une définition du gestionnaire de files d'attente QM1 vers le gestionnaire de files d'attente QM2.

Procédure

1. Créez un doublon de la définition de rubrique de cluster sur QM2.

Pour le routage direct, définissez tous les attributs pour qu'ils correspondent à la définition de QM1.

Pour le routage via un hôte de rubrique, définissez initialement le nouvel hôte de rubrique en tant que PUB (DISABLED). Cela permet à QM2 de connaître les abonnements dans le cluster, mais pas de démarrer le routage des publications.

2. Attendez que les informations soient propagées via le cluster.

Attendez que la nouvelle définition de rubrique de cluster soit propagée par les gestionnaires de files d'attente de référentiel complet à tous les gestionnaires de files d'attente du cluster. La commande **DISPLAY CLUSTER** permet d'afficher les rubriques de cluster sur chaque membre de cluster et de rechercher une définition provenant de QM2.

Pour le routage de l'hôte de rubrique, attendez que le nouvel hôte de rubrique sur QM2 soit au fait de tous les abonnements. Comparez les abonnements de proxy connus de QM2 et ceux connus de QM1. Pour afficher les abonnements de proxy sur un gestionnaire de files d'attente, vous pouvez exécuter la commande **runmqsc** suivante:

```
DISPLAY SUB(*) SUBTYPE(PROXY)
```

3. Pour le routage via un hôte de rubrique, redéfinissez l'hôte de rubrique sur QM2 en tant que PUB (ENABLED), puis redéfinissez l'hôte de rubrique sur QM1 en tant que PUB (DISABLED).

Maintenant que le nouvel hôte de rubrique sur QM2 a pris connaissance de tous les abonnements sur d'autres gestionnaires de files d'attente, l'hôte de rubrique peut démarrer le routage des publications.

En utilisant le paramètre PUB (DISABLED) pour mettre au repos le trafic des messages via QM1, vous assurez qu'aucune publication n'est en train via QM1 lorsque vous supprimez la définition de rubrique de cluster.

4. Supprimez la définition de rubrique de cluster de QM1.

Vous ne pouvez supprimer la définition de QM1 que si le gestionnaire de files d'attente est disponible. Sinon, vous devez exécuter les deux définitions jusqu'à ce que QM1 soit redémarré ou supprimé de force.

Si QM1 reste indisponible pendant une longue période et que vous devez modifier la définition de rubrique en cluster sur QM2, la définition QM2 est plus récente que la définition QM1 et prévaut donc généralement.

Au cours de cette période, s'il existe des différences entre les définitions sur QM1 et QM2, les erreurs sont consignées dans les journaux d'erreurs des deux gestionnaires de files d'attente, vous avertissant de la définition de rubrique de cluster en conflit.

Si QM1 ne sera jamais renvoyé au cluster, par exemple en raison d'une mise hors service inattendue suite à une panne matérielle, vous pouvez utiliser la commande **RESET CLUSTER** en dernier recours pour forcer l'éjection du gestionnaire de files d'attente. **RESET CLUSTER** supprime automatiquement tous les objets de rubrique hébergés sur le gestionnaire de files d'attente cible.

Concepts associés

Combinaison des portées de publication et d'abonnement

A partir de IBM WebSphere MQ 7.0 , la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

Combinaison d'espaces de sujet dans des réseaux de publication / abonnement

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

Tâches associées

Configuration d'un cluster de publication / abonnement

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER** . Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLROUTE** .

Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement

Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

Avant de commencer

La définition du même objet de rubrique de cluster sur plusieurs gestionnaires de files d'attente n'est fonctionnellement utile que pour un cluster routé par un hôte de rubrique. La définition de plusieurs rubriques correspondantes dans un cluster routé directement ne modifie pas son comportement. Cette tâche s'applique uniquement aux clusters routés par l'hôte de rubrique.

Cette tâche suppose que vous avez lu l'article [Plusieurs définitions de rubrique de cluster portant le même nom](#), en particulier les sections suivantes:

- [Définitions de rubrique de cluster multiples dans un cluster routé via un hôte de rubrique](#)
- [Traitement spécial pour le paramètre PUB](#)

Pourquoi et quand exécuter cette tâche

Lorsqu'un gestionnaire de files d'attente devient un hôte de rubrique routé, il doit d'abord connaître l'existence de toutes les rubriques associées auxquelles il a été abonné dans le cluster. Si des publications sont publiées dans ces rubriques lors de l'ajout d'un hôte de rubrique supplémentaire

et qu'une publication est acheminée vers le nouvel hôte avant que cet hôte n'ait appris l'existence d'abonnements sur d'autres gestionnaires de files d'attente dans le cluster, le nouvel hôte ne transmet pas cette publication à ces abonnements. Cela entraîne l'échec des abonnements aux publications.

Les publications ne sont pas acheminées via des gestionnaires de files d'attente d'hôte de rubrique qui ont explicitement défini le paramètre **PUB** de l'objet de rubrique de cluster sur **DISABLED**. Vous pouvez donc utiliser ce paramètre pour vous assurer qu'aucun abonnement ne manque de publications lors du processus d'ajout d'un hôte de rubrique supplémentaire.

Remarque : Alors qu'un gestionnaire de files d'attente héberge une rubrique de cluster qui a été définie comme **PUB (DISABLED)**, les diffuseurs connectés à ce gestionnaire de files d'attente ne peuvent pas publier de messages et les abonnements correspondants sur ce gestionnaire de files d'attente ne reçoivent pas les publications publiées sur les autres gestionnaires de files d'attente du cluster. Pour cette raison, vous devez prendre en compte la définition de rubriques routées par l'hôte de rubrique sur les gestionnaires de files d'attente où des abonnements existent et où les applications de publication se connectent.

Procédure

1. Configurez un nouvel hôte de rubrique et définissez initialement le nouvel hôte de rubrique en tant que **PUB (DISABLED)**.

Cela permet au nouvel hôte de rubrique de connaître les abonnements dans le cluster, mais pas de démarrer le routage des publications.

Pour plus d'informations sur la configuration d'un hôte de rubrique, voir [«Configuration d'un cluster de publication / abonnement»](#), à la page 474.

2. Déterminez quand le nouvel hôte de rubrique a pris connaissance de tous les abonnements.

Pour ce faire, comparez les abonnements de proxy connus du nouvel hôte de rubrique et ceux connus de l'hôte de rubrique existant. Une façon d'afficher les abonnements de proxy consiste à émettre la commande **runmqsc** suivante: **DISPLAY SUB(*) SUBTYPE (PROXY)**

3. Redéfinissez le nouvel hôte de rubrique en tant que **PUB (ENABLED)**.

Une fois que le nouvel hôte de rubrique a pris connaissance de tous les abonnements des autres gestionnaires de files d'attente, la rubrique peut démarrer le routage des publications.

Concepts associés

[Combinaison des portées de publication et d'abonnement](#)

A partir de IBM WebSphere MQ 7.0, la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

[Combinaison d'espaces de sujet dans des réseaux de publication / abonnement](#)

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

Tâches associées

[Configuration d'un cluster de publication / abonnement](#)

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER**. Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLROUTE**.

[Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente](#)

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

[Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement](#)

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement
Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

Combinaison des portées de publication et d'abonnement

A partir de IBM WebSphere MQ 7.0, la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

Les publications peuvent être diffusées vers tous les gestionnaires de files d'attente connectés dans une topologie de publication / abonnement, ou uniquement vers le gestionnaire de files d'attente local. De même pour les abonnements de proxy. Les publications qui correspondent à un abonnement sont régies par la combinaison de ces deux flux.

La portée des publications et des abonnements peut être QMGR ou ALL. Si un diffuseur de publications et un abonné sont tous deux connectés au même gestionnaire de files d'attente, les paramètres de portée n'affectent pas les publications que l'abonné reçoit de ce diffuseur.

Si le diffuseur de publications et l'abonné sont connectés à des gestionnaires de files d'attente différents, les deux paramètres doivent avoir la valeur ALL pour recevoir les publications distantes.

Supposons que les diffuseurs de publications soient connectés à des gestionnaires de files d'attente différents. Si vous souhaitez qu'un abonné reçoive des publications d'un diffuseur de publications, définissez la portée de l'abonnement sur ALL. Vous pouvez ensuite décider, pour chaque diffuseur de publications, de limiter la portée de ses publications aux abonnés locaux du diffuseur de publications.

Supposons que les abonnés soient connectés à des gestionnaires de files d'attente différents. Si vous souhaitez que les publications d'un diffuseur de publications soient envoyées à tous les abonnés, définissez la portée de la publication sur ALL. Si vous souhaitez qu'un abonné reçoive des publications uniquement d'un diffuseur de publications connecté au même gestionnaire de files d'attente, définissez la portée de l'abonnement sur QMGR.

Exemple: service de résultats de football

Supposons que vous êtes une équipe membre d'une ligue de football. Chaque équipe dispose d'un gestionnaire de files d'attente connecté à toutes les autres équipes d'un cluster de publication / abonnement.

Les équipes publient les résultats de tous les matchs joués sur leur terrain à domicile en utilisant le sujet, `Football/result/Home team name/Away team name`. Les chaînes en italique sont des noms de rubrique variables et la publication est le résultat de la correspondance.

Chaque club republie également les résultats uniquement pour le club à l'aide de la chaîne de rubrique `Football/myteam/Home team name/Away team name`.

Les deux rubriques sont publiées dans l'ensemble du cluster.

Les abonnements suivants ont été mis en place par la ligue afin que les fans de n'importe quelle équipe puissent s'abonner aux résultats de trois manières intéressantes.

Notez que vous pouvez configurer des rubriques de cluster avec SUBSCOPE (QMGR). Les définitions de rubrique sont propagées à chaque membre du cluster, mais la portée de l'abonnement est uniquement le gestionnaire de files d'attente local. Ainsi, les abonnés de chaque gestionnaire de files d'attente reçoivent des publications différentes du même abonnement.

Recevoir tous les résultats

```
DEFINE TOPIC(A) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(ALL)
```

Recevoir tous les résultats de la maison

```
DEFINE TOPIC(B) TOPICSTR('Football/result/') CLUSTER SUBSCOPE(QMGR)
```


Etant donné que l'abonnement possède la portée QMGR , seuls les résultats publiés sur la page d'accueil sont mis en correspondance.

Recevoir tous les résultats de mes équipes

```
DEFINE TOPIC(C) TOPICSTR('Football/myteam/') CLUSTER SUBSCOPE(QMGR)
```

Etant donné que l'abonnement a la portée QMGR , seuls les résultats de l'équipe locale, qui sont republiés localement, sont mis en correspondance.

Concepts associés

Combinaison d'espaces de sujet dans des réseaux de publication / abonnement

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

[Réseaux de publication/abonnement répartis](#)

[Portée de la publication](#)

[Portée de l'abonnement](#)

Tâches associées

[Configuration d'un cluster de publication / abonnement](#)

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER** . Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLRROUTE** .

[Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente](#)

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

[Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique](#)

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

[Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement](#)

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

[Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement](#)

Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

Combinaison d'espaces de sujet dans des réseaux de publication / abonnement

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

Vous pouvez créer différents espaces de sujet de publication / abonnement à l'aide des blocs de construction des attributs **CLUSTER**, **PUBSCOPE** et **SUBSCOPE** , des clusters de publication / abonnement et des hiérarchies de publication / abonnement.

A partir de l'exemple de mise à l'échelle à partir d'un seul gestionnaire de files d'attente vers un cluster de publication / abonnement, les scénarios suivants illustrent différentes topologies de publication / abonnement.

Concepts associés

Combinaison des portées de publication et d'abonnement

A partir de IBM WebSphere MQ 7.0, la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

Réseaux de publication/abonnement répartis

Espaces de sujet

Tâches associées

Configuration d'un cluster de publication / abonnement

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER**. Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLROUTE**.

Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement

Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

Définition de rubriques de cluster

Création d'un espace de sujet unique dans un cluster de publication / abonnement

Mise à l'échelle d'un système de publication / abonnement pour une exécution sur plusieurs gestionnaires de files d'attente. Utilisez un cluster de publication / abonnement pour fournir à chaque diffuseur de publications et abonné un espace de sujet identique unique.

Avant de commencer

Vous avez implémenté un système de publication / abonnement sur un gestionnaire de files d'attente version 7 unique.

Créez toujours des espaces de sujet avec leurs propres rubriques racine, au lieu de dépendre de l'héritage des attributs de SYSTEM.BASE.TOPIC. Si vous mettez à l'échelle votre système de publication / abonnement dans un cluster, vous pouvez définir vos rubriques racine en tant que rubriques de cluster, sur l'hôte de rubrique de cluster, puis toutes vos rubriques sont partagées dans le cluster.

Pourquoi et quand exécuter cette tâche

Vous souhaitez maintenant mettre à l'échelle le système pour qu'il prenne en charge davantage de diffuseurs et d'abonnés et que chaque rubrique soit visible dans le cluster.

Procédure

1. Créez un cluster à utiliser avec le système de publication / abonnement.

Si vous disposez d'un cluster traditionnel existant, pour des raisons de performances, il est préférable de configurer un nouveau cluster pour le nouveau système de publication / abonnement. Vous pouvez utiliser les mêmes serveurs pour les référentiels de cluster des deux clusters

2. Choisissez un gestionnaire de files d'attente, éventuellement l'un des référentiels, comme hôte de rubrique de cluster.
3. Assurez-vous que chaque rubrique qui doit être visible dans le cluster de publication / abonnement est résolue en un objet de rubrique d'administration.
Définissez l'attribut **CLUSTER** en nommant le cluster de publication / abonnement.

Que faire ensuite

Connectez les applications de diffuseur de publications et d'abonné à tous les gestionnaires de files d'attente du cluster.

Créez des objets de rubrique d'administration qui possèdent l'attribut **CLUSTER**. Les rubriques sont également propagées dans le cluster. Les programmes de diffuseur de publications et d'abonné utilisent les rubriques d'administration de sorte que leur comportement ne soit pas modifié en étant connectés à des gestionnaires de files d'attente différents dans le cluster

Si vous avez besoin que SYSTEM.BASE.TOPIC agisse comme une rubrique de cluster sur chaque gestionnaire de files d'attente, vous devez la modifier sur chaque gestionnaire de files d'attente.

Concepts associés

[Réseaux de publication/abonnement répartis](#)

[Espaces de sujet](#)

Tâches associées

[Combinaison des espaces de sujet de plusieurs clusters](#)

Créez des espaces de sujet qui s'étendent sur plusieurs clusters. Publier dans une rubrique d'un cluster et s'y abonner dans un autre.

[Combinaison et isolement d'espaces de sujet dans plusieurs clusters](#)

Isolez certains espaces de sujet dans un cluster spécifique et combinez d'autres espaces de sujet pour les rendre accessibles dans tous les clusters connectés.

[Publication et abonnement à des espaces de sujet dans plusieurs clusters](#)

Publication et abonnement à des rubriques dans plusieurs clusters à l'aide de clusters chevauchés. Vous pouvez utiliser cette technique tant que les espaces de sujet dans les clusters ne se chevauchent pas.

[Définition de rubriques de cluster](#)

Combinaison des espaces de sujet de plusieurs clusters

Créez des espaces de sujet qui s'étendent sur plusieurs clusters. Publier dans une rubrique d'un cluster et s'y abonner dans un autre.

Avant de commencer

Cette tâche suppose que vous disposez déjà de clusters de publication / abonnement routés directement et que vous souhaitez propager certaines rubriques de cluster dans tous les clusters.

Remarque : Vous ne pouvez pas effectuer cette opération pour les clusters de publication / abonnement routés par l'hôte de rubrique.

Pourquoi et quand exécuter cette tâche

Pour propager des publications d'un cluster à un autre, vous devez joindre les clusters dans une hiérarchie ; voir [Figure 65](#), à la [page 484](#). Les connexions hiérarchiques propagent les abonnements et les publications entre les gestionnaires de files d'attente connectés et les clusters propagent les rubriques de cluster dans chaque cluster, mais pas entre les clusters.

La combinaison de ces deux mécanismes propage les rubriques de cluster entre tous les clusters. Vous devez répéter les définitions de rubrique de cluster dans chaque cluster.

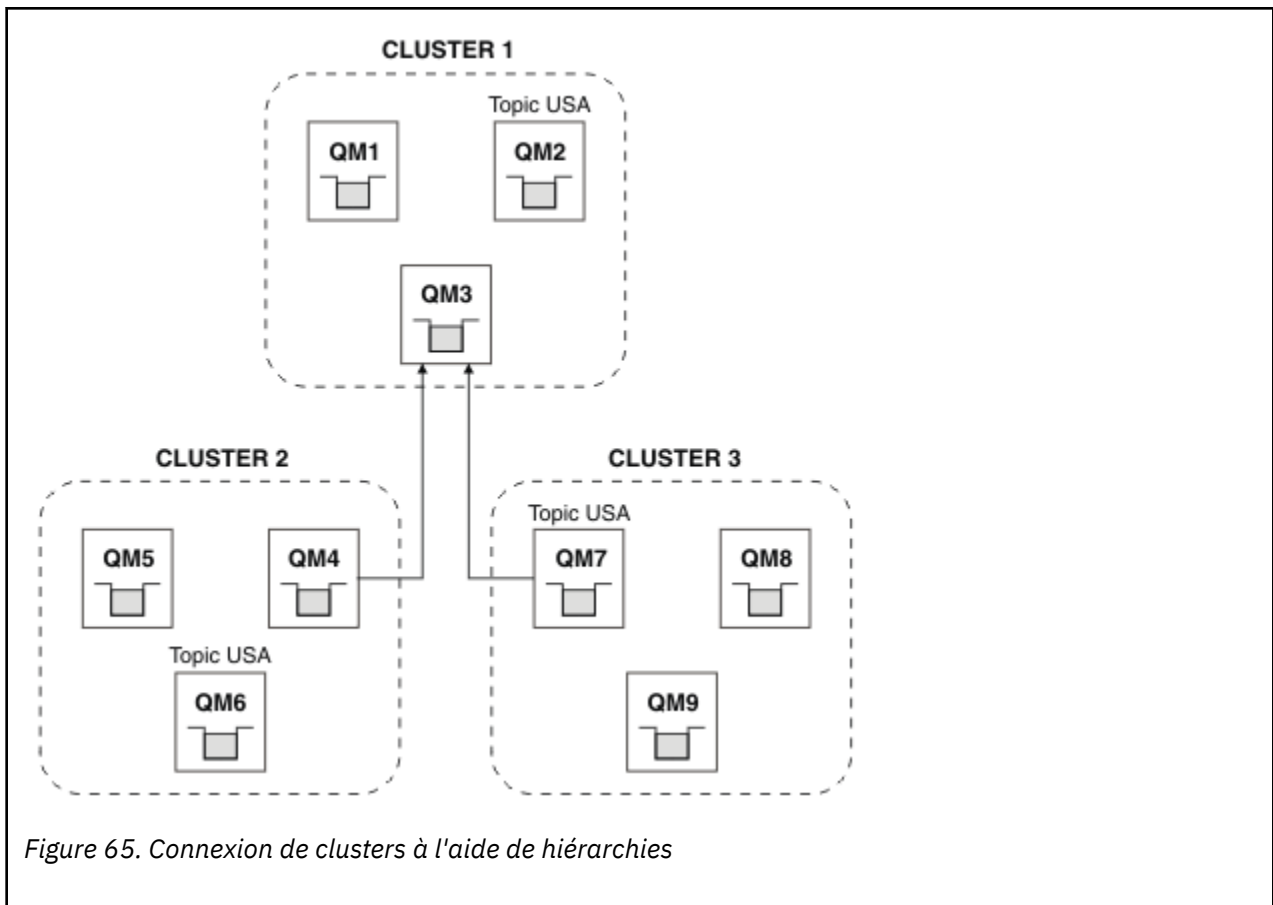


Figure 65. Connexion de clusters à l'aide de hiérarchies

Les étapes suivantes permettent de connecter les clusters dans une hiérarchie.

Procédure

1. Créez deux ensembles de canaux émetteur-récepteur pour connecter QM3 et QM4, et QM3 et QM7, dans les deux sens. Vous devez utiliser des canaux émetteurs-récepteurs traditionnels et des files d'attente de transmission, plutôt qu'un cluster, pour connecter une hiérarchie.
2. Créez trois files d'attente de transmission avec les noms des gestionnaires de files d'attente cible. Utilisez des alias de gestionnaire de files d'attente si vous ne pouvez pas utiliser le nom du gestionnaire de files d'attente cible comme nom de file d'attente de transmission pour une raison quelconque.
3. Configurez les files d'attente de transmission pour déclencher les canaux émetteurs.
4. Vérifiez que le **PSMODE** de QM3, QM4 et QM7 est défini sur ENABLE.
5. Modifiez l'attribut **PARENT** de QM4 et QM7 en QM3.
6. Vérifiez que le statut de la relation parent-enfant entre les gestionnaires de files d'attente est actif dans les deux sens.
7. Créez la rubrique d'administration USA avec l'attribut **CLUSTER** (' CLUSTER 1 '), **CLUSTER** (' CLUSTER 2 ') et **CLUSTER** (' CLUSTER 3 ') sur chacun des trois gestionnaires de files d'attente d'hôte de rubrique de cluster dans les clusters 1, 2 et 3. L'hôte de rubrique de cluster n'a pas besoin d'être un gestionnaire de files d'attente connecté de manière hiérarchique.

Que faire ensuite

Vous pouvez désormais publier ou vous abonner à la rubrique de cluster USA dans [Figure 65](#), à la page 484. Les abonnements aux publications sont transmis aux diffuseurs de publications et aux abonnés dans les trois clusters.

Supposons que vous n'avez pas créé USA en tant que rubrique de cluster dans les autres clusters. Si USA est défini uniquement sur QM7, les publications et les abonnements à USA sont échangés entre QM7, QM8, QM9 et QM3. Les diffuseurs de publications et les abonnés s'exécutant sous QM7, QM8, QM9 héritent des attributs de la rubrique d'administration USA. Les diffuseurs de publications et les abonnés sur QM3 héritent des attributs de SYSTEM.BASE.TOPIC sur QM3.

Voir aussi [«Combinaison et isolement d'espaces de sujet dans plusieurs clusters»](#), à la page 485.

Concepts associés

[Réseaux de publication/abonnement répartis](#)

[Espaces de sujet](#)

Tâches associées

[Création d'un espace de sujet unique dans un cluster de publication / abonnement](#)

Mise à l'échelle d'un système de publication / abonnement pour une exécution sur plusieurs gestionnaires de files d'attente. Utilisez un cluster de publication / abonnement pour fournir à chaque diffuseur de publications et abonné un espace de sujet identique unique.

[Combinaison et isolement d'espaces de sujet dans plusieurs clusters](#)

Isolez certains espaces de sujet dans un cluster spécifique et combinez d'autres espaces de sujet pour les rendre accessibles dans tous les clusters connectés.

[Publication et abonnement à des espaces de sujet dans plusieurs clusters](#)

Publication et abonnement à des rubriques dans plusieurs clusters à l'aide de clusters chevauchés. Vous pouvez utiliser cette technique tant que les espaces de sujet dans les clusters ne se chevauchent pas.

[Définition de rubriques de cluster](#)

Combinaison et isolement d'espaces de sujet dans plusieurs clusters

Isolez certains espaces de sujet dans un cluster spécifique et combinez d'autres espaces de sujet pour les rendre accessibles dans tous les clusters connectés.

Avant de commencer

Consultez la rubrique [«Combinaison des espaces de sujet de plusieurs clusters»](#), à la page 483. Cela peut être suffisant pour vos besoins, sans ajouter de gestionnaire de files d'attente supplémentaire en tant que pont.

Remarque : Vous ne pouvez effectuer cette tâche qu'à l'aide de clusters de publication / abonnement routés directement. Vous ne pouvez pas effectuer cette opération à l'aide de clusters routés par l'hôte de rubrique.

Pourquoi et quand exécuter cette tâche

Une amélioration potentielle de la topologie affichée dans [Figure 65](#), à la page 484 dans [«Combinaison des espaces de sujet de plusieurs clusters»](#), à la page 483 consiste à isoler les rubriques de cluster qui ne sont pas partagées entre tous les clusters. Isolez les clusters en créant un gestionnaire de files d'attente de pontage qui ne se trouve dans aucun des clusters ; voir [Figure 66](#), à la page 486. Utilisez le gestionnaire de files d'attente de pontage pour filtrer les publications et les abonnements qui peuvent être transmis d'un cluster à un autre.

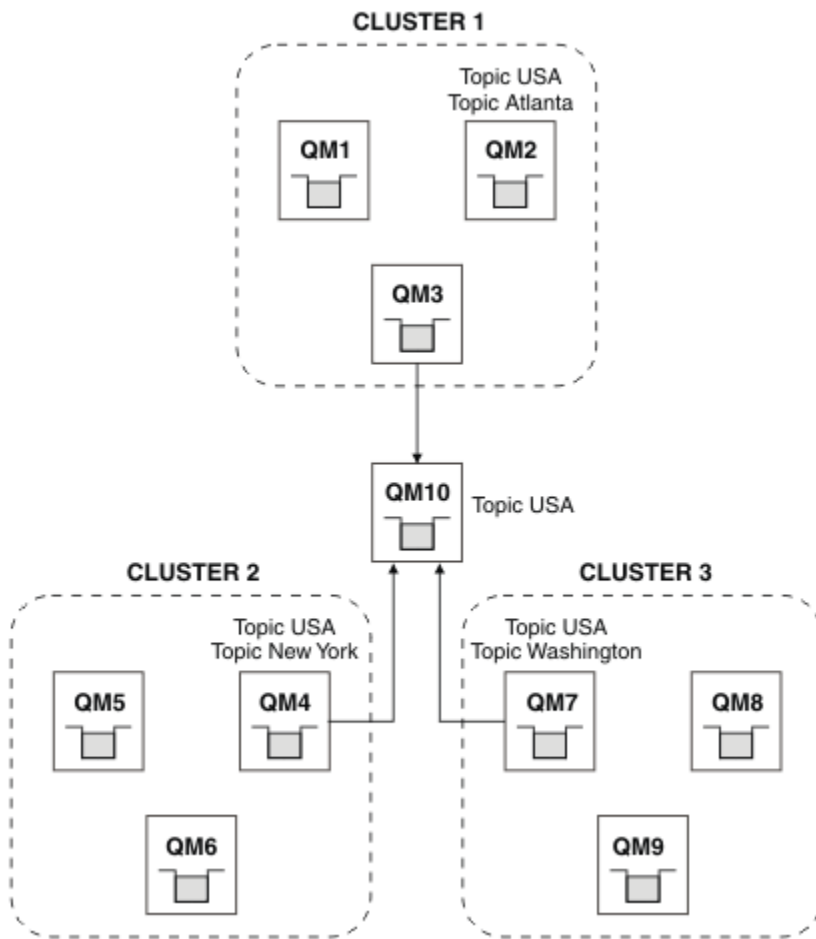


Figure 66. Clusters pontés

Utilisez le pont pour isoler les rubriques de cluster que vous ne souhaitez pas exposer sur le pont sur les autres clusters. Dans [Figure 66](#), à la page 486, USA est une rubrique de cluster partagée dans tous les clusters, et Atlanta, New York et Washington sont des rubriques de cluster qui ne sont partagées que dans un seul cluster.

Modélisez votre configuration à l'aide de la procédure suivante:

Procédure

1. Modifiez tous les objets de rubrique SYSTEM.BASE.TOPIC pour qu'ils aient **SUBSCOPE** (QMGR) et **PUBSCOPE** (QMGR) sur tous les gestionnaires de files d'attente.
Aucune rubrique (même les rubriques de cluster) n'est propagée à d'autres gestionnaires de files d'attente sauf si vous définissez explicitement **SUBSCOPE** (ALL) et **PUBSCOPE** (ALL) sur la rubrique racine de vos rubriques de cluster.
2. Définissez les rubriques sur les trois gestionnaires de files d'attente d'hôte de rubrique de cluster que vous souhaitez partager dans chaque cluster avec les attributs **CLUSTER** (*clustername*), **SUBSCOPE** (ALL) et **PUBSCOPE** (ALL).
Si vous souhaitez que certaines rubriques de cluster soient partagées entre tous les clusters, définissez la même rubrique dans chacun des clusters. Utilisez le nom de chaque cluster comme attribut de cluster.
3. Pour les rubriques de cluster que vous souhaitez partager entre tous les clusters, définissez à nouveau les rubriques sur le gestionnaire de files d'attente de pont (QM10), avec les attributs **SUBSCOPE** (ALL) et **PUBSCOPE** (ALL).

Exemple

Dans l'exemple de la [Figure 66](#), à la page 486, seules les rubriques qui héritent de USA se propagent entre les trois clusters.

Que faire ensuite

Abonnements pour les rubriques définies sur le gestionnaire de files d'attente de pont avec **SUBSCOPE** (ALL) et **PUBSCOPE** (ALL) sont propagées entre les clusters.

Abonnements aux rubriques définies dans chaque cluster avec les attributs **CLUSTER** (*nom_cluster*), **SUBSCOPE** (ALL) et **PUBSCOPE** (ALL) sont propagées dans chaque cluster.

Tous les autres abonnements sont locaux à un gestionnaire de files d'attente.

Concepts associés

[Réseaux de publication/abonnement répartis](#)

[Espaces de sujet](#)

[Portée de la publication](#)

[Portée de l'abonnement](#)

Tâches associées

[Création d'un espace de sujet unique dans un cluster de publication / abonnement](#)

Mise à l'échelle d'un système de publication / abonnement pour une exécution sur plusieurs gestionnaires de files d'attente. Utilisez un cluster de publication / abonnement pour fournir à chaque diffuseur de publications et abonné un espace de sujet identique unique.

[Combinaison des espaces de sujet de plusieurs clusters](#)

Créez des espaces de sujet qui s'étendent sur plusieurs clusters. Publier dans une rubrique d'un cluster et s'y abonner dans un autre.

[Publication et abonnement à des espaces de sujet dans plusieurs clusters](#)

Publication et abonnement à des rubriques dans plusieurs clusters à l'aide de clusters chevauchés. Vous pouvez utiliser cette technique tant que les espaces de sujet dans les clusters ne se chevauchent pas.

[Définition de rubriques de cluster](#)

Publication et abonnement à des espaces de sujet dans plusieurs clusters

Publication et abonnement à des rubriques dans plusieurs clusters à l'aide de clusters chevauchés. Vous pouvez utiliser cette technique tant que les espaces de sujet dans les clusters ne se chevauchent pas.

Avant de commencer

Créez plusieurs clusters traditionnels avec des gestionnaires de files d'attente dans les intersections entre les clusters.

Pourquoi et quand exécuter cette tâche

Vous avez peut-être choisi de chevaucher des clusters pour différentes raisons.

1. Vous disposez d'un nombre limité de serveurs à haute disponibilité ou de gestionnaires de files d'attente. Vous décidez de déployer tous les référentiels de cluster et les hôtes de rubrique de cluster sur ces derniers.
2. Vous disposez de clusters de gestionnaires de files d'attente traditionnels qui sont connectés à l'aide de gestionnaires de files d'attente de passerelle. Vous souhaitez déployer des applications de publication / abonnement dans la même topologie de cluster.
3. Vous disposez de plusieurs applications de publication / abonnement autonomes. Pour des raisons de performances, il est préférable de conserver les clusters de publication / abonnement petits et séparés des clusters traditionnels. Vous avez décidé de déployer les applications dans différents clusters. Toutefois, vous souhaitez également surveiller toutes les applications de publication / abonnement sur un gestionnaire de files d'attente, car vous ne disposez que d'une seule copie de

l'application de surveillance. Ce gestionnaire de files d'attente doit avoir accès aux publications pour les rubriques de cluster dans tous les clusters.

En vous assurant que vos rubriques sont définies dans des espaces de sujet qui ne se chevauchent pas, vous pouvez les déployer dans des clusters de publication / abonnement qui se chevauchent. Voir [Figure 67](#), à la page 488. Si les espaces de sujet se chevauchent, le déploiement dans des clusters qui se chevauchent entraîne des problèmes.

Etant donné que les clusters de publication / abonnement se chevauchent, vous pouvez publier et vous abonner à n'importe quel espace de sujet à l'aide des gestionnaires de files d'attente du chevauchement.

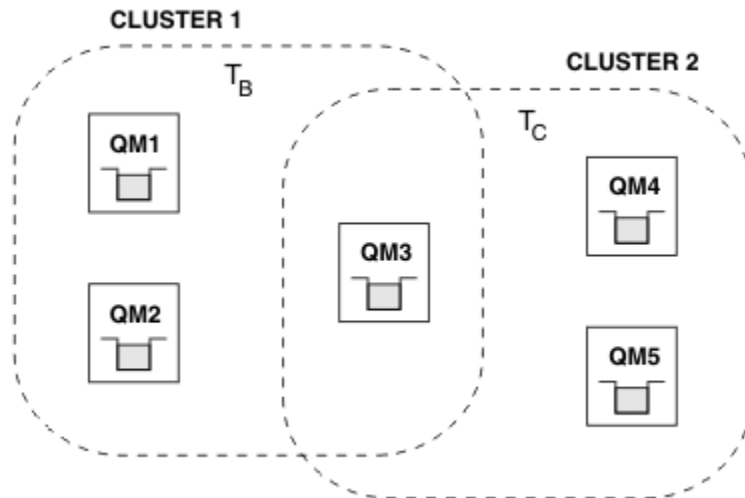


Figure 67. Clusters qui se chevauchent, espaces de sujet qui ne se chevauchent pas

Procédure

Créez un moyen de vous assurer que les espaces de sujet ne se chevauchent pas.

Par exemple, définissez une rubrique racine unique pour chacun des espaces de rubrique. Créez les rubriques de cluster de rubriques racine.

- a) DEFINE TOPIC(B) TOPICSTR('B') CLUSTER('CLUSTER 1') ...
- b) DEFINE TOPIC(C) TOPICSTR('C') CLUSTER('CLUSTER 2') ...

Exemple

Dans [Figure 67](#), à la page 488, les diffuseurs de publications et les abonnés connectés à QM3 peuvent publier ou s'abonner à T_B ou à T_C .

Que faire ensuite

Connectez les diffuseurs de publications et les abonnés qui utilisent des rubriques dans les deux clusters aux gestionnaires de files d'attente dans le chevauchement.

Connectez les diffuseurs de publications et les abonnés qui ne doivent utiliser que les rubriques d'un cluster spécifique aux gestionnaires de files d'attente qui ne se trouvent pas dans le chevauchement.

Concepts associés

[Réseaux de publication/abonnement répartis](#)

[Espaces de sujet](#)

Tâches associées

[Création d'un espace de sujet unique dans un cluster de publication / abonnement](#)

Mise à l'échelle d'un système de publication / abonnement pour une exécution sur plusieurs gestionnaires de files d'attente. Utilisez un cluster de publication / abonnement pour fournir à chaque diffuseur de publications et abonné un espace de sujet identique unique.

Combinaison des espaces de sujet de plusieurs clusters

Créez des espaces de sujet qui s'étendent sur plusieurs clusters. Publier dans une rubrique d'un cluster et s'y abonner dans un autre.

Combinaison et isolement d'espaces de sujet dans plusieurs clusters

Isolez certains espaces de sujet dans un cluster spécifique et combinez d'autres espaces de sujet pour les rendre accessibles dans tous les clusters connectés.

Définition de rubriques de cluster

Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

Avant de commencer

1. Les gestionnaires de files d'attente d'une hiérarchie de publication / abonnement doivent avoir des noms de gestionnaire de files d'attente uniques.
2. Une hiérarchie de publication / abonnement repose sur la fonction de gestionnaire de files d'attente de "publication / abonnement en file d'attente" . Cette option doit être activée à la fois sur le gestionnaire de files d'attente parent et sur le gestionnaire de files d'attente enfant. Voir [«Démarrage de la publication/l'abonnement en file d'attente»](#), à la page 470.
3. La relation de publication / abonnement repose sur les canaux émetteur et récepteur du gestionnaire de files d'attente. Il existe deux façons d'établir les canaux:
 - Ajoutez les gestionnaires de files d'attente parent et enfant à un cluster IBM MQ . Voir [«Ajout d'un gestionnaire de files d'attente à un cluster»](#), à la page 343.
 - Etablissez une paire de canaux émetteur / récepteur du gestionnaire de files d'attente enfant vers le parent et du parent vers l'enfant. Chaque canal doit utiliser une file d'attente de transmission portant le même nom que le gestionnaire de files d'attente cible ou un alias de gestionnaire de files d'attente portant le même nom que le gestionnaire de files d'attente cible. Pour plus d'informations sur l'établissement d'une connexion de canal point à point, voir [«Techniques de mise en file d'attente répartie IBM MQ»](#), à la page 213.

Pour des exemples qui configurent une hiérarchie sur chaque type de configuration de canal, voir l'ensemble suivant de scénarios de hiérarchie de publication / abonnement:

- [Scénario 1 : Utilisation des canaux point-à-point avec un alias de nom de gestionnaire de files d'attente](#)
- [Scénario 2 : Utilisation des canaux point-à-point pour lesquels le nom de la file d'attente de transmission est le même que le nom du gestionnaire de files d'attente éloignées](#)
- [Scénario 3 : Utilisation d'un canal de cluster pour ajouter un gestionnaire de files d'attente](#)

Pourquoi et quand exécuter cette tâche

Utilisez la commande ALTER QMGR PARENT (*PARENT_NAME*) **runmqsc** pour connecter les enfants aux parents. Cette configuration est effectuée sur le gestionnaire de files d'attente enfant, où *PARENT_NAME* est le nom du gestionnaire de files d'attente parent.

Procédure

```
ALTER QMGR PARENT (NOM_PARE)
```

Exemple

Le premier exemple montre comment connecter le gestionnaire de files d'attente QM2 en tant qu'enfant de QM1, puis interroger QM2 pour confirmer qu'il est devenu un enfant avec un **STATUS** de ACTIVE:

```
C:>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2
alter qmgr parent(QM1)
  1 : alter qmgr parent(QM1)
AMQ8005: IBM MQ queue manager changed.
display pubsub all
  2 : display pubsub all
AMQ8723: Display pub/sub status details.
      QMNAME(QM2)                TYPE(LOCAL)
      STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
      QMNAME(QM1)                TYPE(PARENT)
      STATUS(ACTIVE)
```

L'exemple suivant montre le résultat de l'interrogation de QM1 pour ses connexions:

```
C:\Documents and Settings\Admin>runmqsc QM1
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM1.
display pubsub all
  2 : display pubsub all
AMQ8723: Display pub/sub status details.
      QMNAME(QM1)                TYPE(LOCAL)
      STATUS(ACTIVE)
AMQ8723: Display pub/sub status details.
      QMNAME(QM2)                TYPE(CHILD)
      STATUS(ACTIVE)
```

Si **STATUS** ne s'affiche pas comme ACTIVE, vérifiez que les canaux entre l'enfant et le parent sont correctement configurés et en cours d'exécution. Recherchez les erreurs possibles dans les deux journaux d'erreurs du gestionnaire de files d'attente.

Que faire ensuite

Par défaut, les rubriques utilisées par les diffuseurs et les abonnés d'un gestionnaire de files d'attente sont partagées avec les diffuseurs et les abonnés des autres gestionnaires de files d'attente de la hiérarchie. Les rubriques administrées peuvent être configurées pour contrôler le niveau de partage via l'utilisation des propriétés de rubrique **SUBSCOPE** et **PUBSCOPE**. Voir [«Configuration des réseaux de publication/abonnement réparti»](#), à la page 473.

Concepts associés

[Combinaison des portées de publication et d'abonnement](#)

A partir de IBM WebSphere MQ 7.0, la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

[Combinaison d'espaces de sujet dans des réseaux de publication / abonnement](#)

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

Tâches associées

[Configuration d'un cluster de publication / abonnement](#)

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER**. Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLROUTE**.

[Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente](#)

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement

Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

Référence associée

Flux et rubriques

AFFICHAGE DE PUBSUB

Messagerie de type publication/abonnement

Déconnexion d'un gestionnaire de files d'attente d'une hiérarchie de publication / abonnement

Déconnectez un gestionnaire de files d'attente enfant d'un gestionnaire de files d'attente parent dans une hiérarchie de publication / abonnement.

Pourquoi et quand exécuter cette tâche

La commande **ALTER QMGR** permet de déconnecter un gestionnaire de files d'attente d'une hiérarchie de courtiers. Vous pouvez déconnecter un gestionnaire de files d'attente dans n'importe quel ordre à tout moment.

La demande correspondante pour la mise à jour du parent est envoyée lorsque la connexion entre les gestionnaires de files d'attente est en cours d'exécution.

Procédure

```
ALTER QMGR PARENT( '')
```

Exemple

```
C:\Documents and Settings\Admin>runmqsc QM2
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.
Starting MQSC for queue manager QM2.
 1 : alter qmgr parent('')
AMQ8005: IBM MQ queue manager changed.
 2 : display pubsub type(child)
AMQ8147: IBM MQ object not found.
display pubsub type(parent)
 3 : display pubsub type(parent)
AMQ8147: IBM MQ object not found.
```

Que faire ensuite

Vous pouvez supprimer tous les flux, files d'attente et canaux définis manuellement qui ne sont plus nécessaires.

Concepts associés

Combinaison des portées de publication et d'abonnement

A partir de IBM WebSphere MQ 7.0, la portée des publications et des abonnements fonctionne indépendamment pour déterminer le flux des publications entre les gestionnaires de files d'attente.

Combinaison d'espaces de sujet dans des réseaux de publication / abonnement

Combinez l'espace de sujet d'un gestionnaire de files d'attente avec d'autres gestionnaires de files d'attente dans un cluster ou une hiérarchie de publication / abonnement. Combinez des clusters de publication / abonnement et des clusters de publication / abonnement avec des hiérarchies.

Tâches associées

Configuration d'un cluster de publication / abonnement

Définissez une rubrique sur un gestionnaire de files d'attente. Pour faire de la rubrique une rubrique de cluster, définissez la propriété **CLUSTER** . Pour choisir le routage à utiliser pour les publications et les abonnements de cette rubrique, définissez la propriété **CLROUTE** .

Déplacement d'une définition de rubrique de cluster vers un autre gestionnaire de files d'attente

Pour les clusters routés via un hôte de rubrique ou routés directement, il peut être nécessaire de déplacer une définition de rubrique de cluster lors de la mise hors service d'un gestionnaire de files d'attente ou parce qu'un gestionnaire de files d'attente de cluster a échoué ou est indisponible pendant une période importante.

Ajout d'hôtes de rubrique supplémentaires à un cluster routé par un hôte de rubrique

Dans un cluster de publication / abonnement routé par un hôte de rubrique, plusieurs gestionnaires de files d'attente peuvent être utilisés pour acheminer des publications vers des abonnements en définissant le même objet de rubrique en cluster sur ces gestionnaires de files d'attente. Cela peut être utilisé pour améliorer la disponibilité et l'équilibrage de la charge de travail. Lorsque vous ajoutez un hôte de rubrique supplémentaire pour le même objet de rubrique de cluster, vous pouvez utiliser le paramètre **PUB** pour contrôler le moment où les publications commencent à être acheminées via le nouvel hôte de rubrique.

Connexion d'un gestionnaire de files d'attente à une hiérarchie de publication/abonnement

Vous connectez le gestionnaire de files d'attente enfant au gestionnaire de files d'attente parent dans la hiérarchie. Si le gestionnaire de files d'attente enfant est déjà membre d'une autre hiérarchie ou d'un autre cluster, cette connexion relie les hiérarchies ou le cluster à la hiérarchie.

ALW

Configuration de plusieurs installations

Lorsque vous utilisez plusieurs installations sur le même système, vous devez configurer les installations et les gestionnaires de files d'attente.

Pourquoi et quand exécuter cette tâche

Ces informations s'appliquent à AIX, Linux, and Windows.

Procédure

- Utilisez les informations des liens suivants pour configurer vos installations:
 - [«Changement d'installation principale», à la page 500](#)
 - [«Association d'un gestionnaire de files d'attente à une installation», à la page 502](#)
 - [«Connexion d'applications dans un environnement avec plusieurs installations», à la page 492](#)

ALW

Connexion d'applications dans un environnement avec plusieurs installations

Sur les systèmes AIX, Linux, and Windows , si des bibliothèques IBM MQ sont chargées, IBM MQ utilise automatiquement les bibliothèques appropriées sans que vous ayez à effectuer d'autres actions. IBM MQ utilise les bibliothèques de l'installation associée au gestionnaire de files d'attente auquel l'application se connecte.

Les concepts suivants sont utilisés pour expliquer la façon dont les applications se connectent à IBM MQ:

Liaison

Lorsque l'application est compilée, elle est liée aux bibliothèques IBM MQ pour obtenir les exportations de fonction qui sont ensuite chargées lors de l'exécution de l'application.

Chargement

Lorsque l'application est exécutée, les bibliothèques IBM MQ sont localisées et chargées. Le mécanisme spécifique utilisé pour localiser les bibliothèques varie en fonction du système d'exploitation et de la façon dont l'application est générée. Pour plus d'informations sur la façon

de localiser et de charger des bibliothèques dans un environnement à installations multiples, voir [«Chargement des bibliothèques IBM MQ»](#), à la page 494.

Connexion en cours

Lorsque l'application se connecte à un gestionnaire de files d'attente en cours d'exécution, par exemple à l'aide d'un appel MQCONN ou MQCONNX, elle se connecte à l'aide des bibliothèques IBM MQ chargées.

Lorsqu'une application serveur se connecte à un gestionnaire de files d'attente, les bibliothèques chargées doivent provenir de l'installation associée au gestionnaire de files d'attente. Avec plusieurs installations sur un système, cette restriction introduit de nouvelles difficultés lors du choix du mécanisme utilisé par le système d'exploitation pour localiser les bibliothèques IBM MQ à charger:

- Lorsque la commande **setmqm** est utilisée pour modifier l'installation associée à un gestionnaire de files d'attente, les bibliothèques à charger changent.
- Lorsqu'une application se connecte à plusieurs gestionnaires de files d'attente appartenant à des installations différentes, plusieurs ensembles de bibliothèques doivent être chargés.

Toutefois, si IBM MQ, des bibliothèques, sont localisées et chargées, IBM MQ charge et utilise les bibliothèques appropriées sans que vous ayez à effectuer d'autres actions. Lorsque l'application se connecte à un gestionnaire de files d'attente, IBM MQ charge les bibliothèques à partir de l'installation à laquelle le gestionnaire de files d'attente est associé.

Les scénarios de migration et de connexion d'applications avec plusieurs installations sont pris en compte plus en détail dans la rubrique [Multi-installation queue manager coexistence on AIX, Linux, and Windows](#).

Pour plus d'informations sur le chargement des bibliothèques IBM MQ, voir [«Chargement des bibliothèques IBM MQ»](#), à la page 494.

Support et restrictions

Si l'une des bibliothèques IBM MQ suivantes est localisée et chargée, le produit peut automatiquement charger et utiliser les bibliothèques appropriées:

- Les bibliothèques du serveur C
- Les bibliothèques du serveur C++
- Les bibliothèques du serveur XA
- Les bibliothèques du serveur COBOL
- Les bibliothèques du serveur COM +
- .NET en mode non géré

IBM MQ charge et utilise automatiquement les bibliothèques appropriées pour les applications Java et JMS en mode liaisons.

Il existe un certain nombre de restrictions pour les applications utilisant plusieurs installations. Pour plus d'informations, voir [«Restrictions pour les applications qui utilisent plusieurs installations»](#), à la page 497.

Concepts associés

[«Restrictions pour les applications qui utilisent plusieurs installations»](#), à la page 497

Il existe des restrictions quant à l'utilisation des bibliothèques de serveur CICS, des connexions avec accès rapide, des descripteurs de messages et des exits dans un environnement avec plusieurs installations.

[«Chargement des bibliothèques IBM MQ»](#), à la page 494

Lorsque vous décidez comment charger des bibliothèques IBM MQ, vous devez prendre en compte un certain nombre de facteurs, notamment: votre environnement, si vous pouvez modifier vos applications existantes, si vous souhaitez une installation principale, où IBM MQ est installé et si l'emplacement de IBM MQ est susceptible de changer.

Tâches associées

[Choix d'une installation principale](#)

«Changement d'installation principale», à la page 500

Vous pouvez utiliser la commande **setmqinst** pour définir ou annuler la définition d'une installation en tant qu'installation principale.

«Association d'un gestionnaire de files d'attente à une installation», à la page 502

Lorsque vous créez un gestionnaire de files d'attente, il est automatiquement associé à l'installation qui a émis la commande **crtmqm**. Sous AIX, Linux, and Windows, vous pouvez modifier l'installation associée à un gestionnaire de files d'attente à l'aide de la commande **setmqm**.

Chargement des bibliothèques IBM MQ

Lorsque vous décidez comment charger des bibliothèques IBM MQ, vous devez prendre en compte un certain nombre de facteurs, notamment: votre environnement, si vous pouvez modifier vos applications existantes, si vous souhaitez une installation principale, où IBM MQ est installé et si l'emplacement de IBM MQ est susceptible de changer.

La façon dont les bibliothèques IBM MQ sont localisées et chargées dépend de votre environnement d'installation:

- Sur les systèmes AIX and Linux, si une copie d'une version de IBM MQ est installée à l'emplacement par défaut, les applications existantes continuent de fonctionner de la même manière que les versions précédentes. Toutefois, si les applications ont besoin de liens symboliques dans `/usr/lib`, vous devez soit sélectionner une installation de version IBM MQ comme installation principale, soit créer manuellement les liens symboliques.
- Si IBM MQ est installé dans un emplacement autre que l'emplacement par défaut, vous devrez peut-être modifier vos applications existantes afin que les bibliothèques appropriées soient chargées.

La façon dont les bibliothèques IBM MQ peuvent être localisées et chargées dépend également de la manière dont les applications existantes sont configurées pour charger les bibliothèques. Pour plus d'informations sur la façon dont les bibliothèques peuvent être chargées, voir «[Mécanismes de chargement des bibliothèques du système d'exploitation](#)», à la page 496.

De manière optimale, vous devez vous assurer que la bibliothèque IBM MQ, qui est chargée par le système d'exploitation, est celle à laquelle le gestionnaire de files d'attente est associé.

Les méthodes de chargement des bibliothèques IBM MQ varient en fonction de la plateforme, et chaque méthode présente des avantages et des inconvénients.



Plateforme	Option	Avantages	Inconvénients
  Systèmes AIX and Linux	Définissez ou modifiez le chemin de recherche de l'environnement d'exécution intégré (RPath) de l'application. Cette option nécessite que vous recompiliez et liez l'application. Pour plus d'informations sur la compilation et la liaison d'applications, voir Génération d'une application procédurale .	<ul style="list-style-type: none">• La portée de la modification est claire.	<ul style="list-style-type: none">• Vous devez être en mesure de recompiler et de lier l'application.• Si l'emplacement de IBM MQ change, vous devez modifier le chemin d'accès aux données (RPath).

Tableau 29. Avantages et inconvénients des options de chargement des bibliothèques (suite)

Plateforme	Option	Avantages	Inconvénients
<p>Systemes AIX and Linux</p>	<p>Définissez la variable d'environnement <code>LD_LIBRARY_PATH</code> à l'aide de <code>setmqenvou crtmmqenv</code>, avec l'option <code>-k</code> ou <code>-l</code>. (</p> <p>AIX Sous AIX, cette variable d'environnement est <code>LIBPATH</code></p>	<ul style="list-style-type: none"> Aucune modification n'est requise pour les applications existantes. Remplace les chemins d'accès imbriqués dans une application. Il est facile de modifier la variable si l'emplacement de IBM MQ change. 	<ul style="list-style-type: none"> Les applications <code>setuid</code> et <code>setgid</code>, ou les applications générées d'une autre manière, peuvent ignorer <code>LD_LIBRARY_PATH</code> pour des raisons de sécurité. Spécifique à l'environnement, doit donc être défini dans chaque environnement dans lequel l'application est exécutée. Impact possible sur les autres applications qui s'appuient sur <code>LD_LIBRARY_PATH</code>. Linux Linux: le compilateur utilisé pour générer l'application peut désactiver l'utilisation de <code>LD_LIBRARY_PATH</code>. Pour plus d'informations, voir Remarques sur la liaison d'exécution pour Linux.
<p>Windows Systemes Windows</p>	<p>Définissez la variable <code>PATH</code> à l'aide de <code>setmqenvou crtmmqenv</code>.</p>	<ul style="list-style-type: none"> Aucune modification n'est requise pour les applications existantes. Il est facile de modifier la variable si l'emplacement de IBM MQ change. 	<ul style="list-style-type: none"> Spécifique à l'environnement, doit donc être défini dans chaque environnement dans lequel l'application est exécutée. Impact possible sur d'autres applications.
<p>ALW Systemes AIX, Linux, and Windows</p>	<p>Définissez l'installation principale sur une installation IBM MQ ou ultérieure. Voir «Changement d'installation principale», à la page 500.</p> <p>Pour plus d'informations sur l'installation principale, voir Choix d'une installation principale.</p>	<ul style="list-style-type: none"> Aucune modification n'est requise pour les applications existantes. Il est facile de modifier l'installation principale si l'emplacement de IBM MQ change. Donne un comportement similaire aux versions précédentes de IBM MQ. 	<ul style="list-style-type: none"> Linux AIX AIX and Linux: ne fonctionne pas si <code>/usr/lib</code> ne se trouve pas dans le chemin de recherche par défaut.

Remarques sur le chargement des bibliothèques pour Linux

Linux

Les applications compilées à l'aide de certaines versions de gcc, par exemple la version 3.2.x, peuvent avoir un chemin d'accès distant intégré qui ne peut pas être remplacé à l'aide de la variable d'environnement `LD_LIBRARY_PATH`. Vous pouvez déterminer si une application est affectée à l'aide de la commande `readelf -d applicationName`. Le chemin d'accès distant ne peut pas être remplacé si le symbole RPATH est présent et que le symbole RUNPATH n'est pas présent.

Mécanismes de chargement des bibliothèques du système d'exploitation

Sur les systèmes Windows, plusieurs répertoires sont explorés pour trouver les bibliothèques:

- Répertoire à partir duquel l'application est chargée.
- Répertoire de travail.
- Les répertoires de la variable d'environnement `PATH`, à la fois la variable globale `PATH` et la variable `PATH` de l'utilisateur en cours.

Linux **AIX** Sur les systèmes AIX and Linux, un certain nombre de méthodes ont pu être utilisées pour localiser les bibliothèques à charger:

- Utilisation de la variable d'environnement `LD_LIBRARY_PATH` (également `LIBPATH` sur AIX). Si cette variable est définie, elle définit un ensemble de répertoires dans lesquels sont recherchées les bibliothèques IBM MQ requises. Si des bibliothèques sont trouvées dans ces répertoires, elles sont utilisées de préférence à celles qui peuvent être trouvées à l'aide des autres méthodes.
- Utilisation d'un chemin de recherche intégré (RPath). L'application peut contenir un ensemble de répertoires permettant de rechercher les bibliothèques IBM MQ. Si la variable `LD_LIBRARY_PATH` n'est pas définie ou si les bibliothèques requises n'ont pas été trouvées à l'aide de la variable, le chemin d'accès au point de reprise est recherché pour les bibliothèques. Si vos applications existantes utilisent un chemin d'accès distant, mais que vous ne pouvez pas recompiler et lier l'application, vous devez installer IBM MQ dans l'emplacement par défaut ou utiliser une autre méthode pour trouver les bibliothèques.
- Utilisation du chemin d'accès à la bibliothèque par défaut. Si les bibliothèques IBM MQ sont introuvables après la recherche de la variable `LD_LIBRARY_PATH` et des emplacements RPath, le chemin d'accès à la bibliothèque par défaut est recherché. Généralement, ce chemin contient `/usr/lib` ou `/usr/lib64`. Si les bibliothèques sont introuvables après la recherche du chemin d'accès à la bibliothèque par défaut, l'application ne parvient pas à démarrer en raison de dépendances manquantes.

Vous pouvez utiliser les mécanismes du système d'exploitation pour déterminer si vos applications possèdent un chemin de recherche intégré. Exemple :

- **AIX** AIX: `dump`
- **Linux** Linux: `readelf`

Concepts associés

«Restrictions pour les applications qui utilisent plusieurs installations», à la page 497

Il existe des restrictions quant à l'utilisation des bibliothèques de serveur CICS, des connexions avec accès rapide, des descripteurs de messages et des exits dans un environnement avec plusieurs installations.

«Connexion d'applications dans un environnement avec plusieurs installations», à la page 492

Sur les systèmes AIX, Linux, and Windows, si des bibliothèques IBM MQ sont chargées, IBM MQ utilise automatiquement les bibliothèques appropriées sans que vous ayez à effectuer d'autres actions. IBM MQ utilise les bibliothèques de l'installation associée au gestionnaire de files d'attente auquel l'application se connecte.

Tâches associées

[Choix d'une installation principale](#)

[«Changement d'installation principale», à la page 500](#)

Vous pouvez utiliser la commande **setmqinst** pour définir ou annuler la définition d'une installation en tant qu'installation principale.

«Association d'un gestionnaire de files d'attente à une installation», à la page 502

Lorsque vous créez un gestionnaire de files d'attente, il est automatiquement associé à l'installation qui a émis la commande **crtmqm**. Sous AIX, Linux, and Windows, vous pouvez modifier l'installation associée à un gestionnaire de files d'attente à l'aide de la commande **setmqm**.

Restrictions pour les applications qui utilisent plusieurs installations

Il existe des restrictions quant à l'utilisation des bibliothèques de serveur CICS, des connexions avec accès rapide, des descripteurs de messages et des exits dans un environnement avec plusieurs installations.

Bibliothèques du serveur CICS

Si vous utilisez les bibliothèques de serveur CICS, IBM MQ ne sélectionne pas automatiquement le niveau de bibliothèque qui vous convient. Vous devez compiler et lier vos applications au niveau de bibliothèque approprié pour le gestionnaire de files d'attente auquel l'application se connecte. Pour plus d'informations, voir [Génération de bibliothèques à utiliser avec TXSeries for Multiplatforms version 5](#).

Descripteurs de messages

Les descripteurs de messages qui utilisent la valeur spéciale de MQHC_UNASSOCIATED_HCONN ont une utilisation limitée avec la première installation chargée dans un processus. Si le descripteur de message ne peut pas être utilisé par une installation particulière, le code anomalie MQRC_HMSG_NOT_AVAILABLE est renvoyé.

Cette restriction concerne les propriétés du message. Vous ne pouvez pas utiliser les descripteurs de messages pour extraire les propriétés du message d'un gestionnaire de files d'attente sur une installation et les attribuer à un gestionnaire sur une autre installation. Pour plus d'informations sur les descripteurs de messages, voir [MQCRTMH - Création d'un descripteur de message](#).

Exits

Dans un environnement à installations multiples, les exits existants doivent être mis à jour pour être utilisés avec les installations IBM MQ. Les exits de conversion de données générés à l'aide de la commande **crtmqcvx** doivent être régénérés à l'aide de la commande mise à jour.

Tous les exits doivent être écrits à l'aide de la structure MQIEP, ne peuvent pas utiliser un chemin d'accès RPATH intégré pour repérer les bibliothèques IBM MQ et ne peuvent pas être liés aux bibliothèques IBM MQ. Pour plus d'informations, voir [Ecriture des exits et des services optionnels sous AIX, Linux, and Windows](#).

Accès rapide

Sur un serveur comportant plusieurs installations, les applications utilisant une connexion rapide à IBM MQ doivent respecter les règles suivantes:

1. Le gestionnaire de files d'attente doit être associé à la même installation que celle à partir de laquelle l'application a chargé les bibliothèques d'exécution d'IBM MQ. L'application ne doit pas utiliser une connexion avec accès rapide à un gestionnaire de files d'attente associé à une autre installation. Une tentative d'établir la connexion entraîne une erreur et le code anomalie MQRC_INSTALLATION_MISMATCH.
2. La connexion avec accès non rapide à un gestionnaire de files d'attente associé à la même installation que celle à partir de laquelle l'application a chargé les bibliothèques d'exécution d'IBM MQ empêche la connexion avec accès rapide à l'application, à moins qu'une des conditions suivantes soit remplie :

- La première connexion de l'application à un gestionnaire de files d'attente associé à la même installation est une connexion avec accès rapide.
 - La variable d'environnement `AMQ_SINGLE_INSTALLATION` est définie.
3. La connexion avec accès non rapide à un gestionnaire de files d'attente associé à une installation IBM MQ n'a aucun effet sur la capacité d'une application à se connecter avec accès rapide.

Avec `AMQ_SINGLE_INSTALLATION` défini, vous pouvez établir une connexion rapide à n'importe quelle connexion à un gestionnaire de files d'attente. Sinon, presque les mêmes restrictions s'appliquent :

- L'installation doit être la même que celle à partir de laquelle les bibliothèques d'exécution d'IBM MQ ont été chargées.
- Chaque connexion sur le même processus doit s'effectuer à la même installation. Si vous tentez de vous connecter à un gestionnaire de files d'attente associé à une autre installation, la connexion échoue avec le code anomalie `MQRC_INSTALLATION_MISMATCH`. Notez qu'avec l'ensemble `AMQ_SINGLE_INSTALLATION`, cette restriction s'applique à toutes les connexions, et pas uniquement aux connexions rapide.
- Connectez uniquement un gestionnaire de files d'attente en utilisant les connexions avec accès rapide.

Référence associée

[MQCONN - Connexion du gestionnaire de files d'attente \(étendue\)](#)

Structure MQIEP

[2583 \(0A17\) \(RC2583\): MQRC_INSTALLATION_MISMATCH](#)

[2587 \(0A1B\) \(RC2587\): MQRC_HMSG_NOT_AVAILABLE](#)

[2590 \(0A1E\) \(RC2590\): MQRC_FASTPATH_NOT_AVAILABLE](#)

Connexion d'applications .NET dans un environnement à installations multiples

Par défaut, les applications utilisent les assemblages .NET de l'installation principale. S'il n'y a pas d'installation principale ou si vous ne souhaitez pas utiliser les assemblages d'installation principale, vous devez mettre à jour le fichier de configuration d'application ou la variable d'environnement `DEVPATH`.

S'il existe une installation principale sur le système, les assemblages .NET et les fichiers de stratégie de cette installation sont enregistrés dans le cache d'assemblage global (GAC). Les assemblages .NET pour toutes les autres installations se trouvent dans le chemin d'installation de chaque installation, mais les assemblages ne sont pas enregistrés dans le cache d'assemblage global (GAC). Par conséquent, par défaut, les applications s'exécutent à l'aide des assemblages .NET de l'installation principale. Vous devez mettre à jour le fichier de configuration d'application si l'un des cas suivants est vérifié:

- Vous n'avez pas d'installation principale.
- Vous ne souhaitez pas que l'application utilise les assemblages d'installation principaux.
- L'installation principale est une version de IBM MQ antérieure à la version avec laquelle l'application a été compilée.

Pour plus d'informations sur la mise à jour du fichier de configuration d'application, voir [«Connexion d'applications .NET à l'aide du fichier de configuration d'application»](#), à la page 498.

Vous devez mettre à jour la variable d'environnement `DEVPATH` dans les cas suivants:

- Vous souhaitez que votre application utilise les assemblages d'une installation non principale, mais l'installation principale est à la même version que l'installation non principale.

Pour plus d'informations sur la mise à jour de la variable `DEVPATH`, voir [«Connexion d'applications .NET à l'aide de DEVPATH»](#), à la page 500.

Connexion d'applications .NET à l'aide du fichier de configuration d'application

Dans le fichier de configuration d'application, vous devez définir différentes balises pour rediriger les applications afin qu'elles utilisent des assemblages qui ne proviennent pas de l'installation principale.

Le tableau suivant présente les modifications spécifiques à apporter au fichier de configuration d'application pour permettre aux applications .NET de se connecter à l'aide d'assemblages particuliers:

<i>Tableau 30. Configuration des applications pour l'utilisation d'assemblages particuliers</i>		
	Applications compilées avec une version antérieure de IBM MQ	Applications compilées avec une version ultérieure de IBM MQ
Pour exécuter une application avec une installation principale IBM MQ de version ultérieure. (assemblages de versions ultérieures dans GAC):	Aucune modification n'est nécessaire	Aucune modification n'est nécessaire
Pour exécuter une application avec une installation principale IBM MQ de version antérieure. (assemblages de versions antérieures dans GAC):	Aucune modification n'est nécessaire	Dans le fichier de configuration de l'application: <ul style="list-style-type: none"> Utilisez la balise <i>bindingRedirect</i> pour indiquer l'utilisation de la version antérieure des assemblages qui se trouvent dans le cache d'assemblage global (GAC)
Pour exécuter une application avec une version ultérieure de l'installation non principale de IBM MQ . (assemblages de versions ultérieures dans le dossier d'installation):	Dans le fichier de configuration de l'application: <ul style="list-style-type: none"> Utilisez la balise <i>codebase</i> pour pointer vers l'emplacement des assemblages de versions plus récentes Utilisez la balise <i>bindingRedirect</i> pour indiquer l'utilisation des assemblages de versions ultérieures 	Dans le fichier de configuration de l'application: <ul style="list-style-type: none"> Utilisez la balise <i>codebase</i> pour pointer vers l'emplacement des assemblages de versions plus récentes
Pour exécuter une application avec une version antérieure de l'installation non principale de IBM MQ . (assemblages de versions antérieures dans le dossier d'installation):	Dans le fichier de configuration de l'application: <ul style="list-style-type: none"> Utilisez la balise <i>codebase</i> pour pointer vers l'emplacement des assemblages de versions antérieures Incluez la balise <i>publisherpolicy Apply=no</i> 	Dans le fichier de configuration de l'application: <ul style="list-style-type: none"> Utilisez la balise <i>codebase</i> pour pointer vers l'emplacement des assemblages de versions antérieures Utilisez la balise <i>bindingRedirect</i> pour indiquer l'utilisation des assemblages de versions antérieures Incluez la balise <i>publisherpolicy Apply=no</i>

Un exemple de fichier de configuration d'application `NonPrimaryRedirect.config` est fourni dans le dossier `MQ_INSTALLATION_PATH\tools\dotnet\samples\base`. Ce fichier peut être modifié avec le chemin d'installation IBM MQ de toute installation non principale. Le fichier peut également être directement inclus dans d'autres fichiers de configuration à l'aide de la balise *linkedConfiguration*. Des exemples sont fournis pour `nmqsget.exe.config` et `nmqsput.exe.config`. Les deux exemples utilisent la balise *linkedConfiguration* et incluent le fichier `NonPrimaryRedirect.config`.

Connexion d'applications .NET à l'aide de DEVPATH

Vous pouvez trouver les assemblages à l'aide de la variable d'environnement *DEVPATH* . Les assemblages spécifiés par la variable *DEVPATH* sont utilisés de préférence à tous les assemblages du cache d'assemblage global (GAC). Pour plus d'informations sur l'utilisation de cette variable, voir la documentation Microsoft appropriée sur *DEVPATH* .

Pour rechercher les assemblages à l'aide de la variable d'environnement *DEVPATH* , vous devez définir la variable *DEVPATH* sur le dossier qui contient les assemblages que vous souhaitez utiliser. Ensuite, vous devez mettre à jour le fichier de configuration d'application et ajouter les informations de configuration d'exécution suivantes:

```
<configuration>
<runtime>
<developmentMode developerInstallation="true" />
</runtime>
</configuration>
```

Concepts associés

«Connexion d'applications dans un environnement avec plusieurs installations», à la page 492

Sur les systèmes AIX, Linux, and Windows , si des bibliothèques IBM MQ sont chargées, IBM MQ utilise automatiquement les bibliothèques appropriées sans que vous ayez à effectuer d'autres actions. IBM MQ utilise les bibliothèques de l'installation associée au gestionnaire de files d'attente auquel l'application se connecte.

[Installations multiples](#)

Tâches associées

[Choix d'une installation principale](#)

[Utilisation .NET](#)

ALW

Changement d'installation principale

Vous pouvez utiliser la commande **setmqinst** pour définir ou annuler la définition d'une installation en tant qu'installation principale.

Pourquoi et quand exécuter cette tâche

Cette tâche s'applique à AIX, Linux, and Windows.

L'installation principale est l'installation à laquelle les emplacements système requis font référence. Pour plus d'informations sur l'installation principale et sur le choix de votre installation principale, voir [Choix d'une installation principale](#).

Windows

Lors du processus d'installation sous Windows, vous pouvez indiquer que l'installation doit être l'installation principale.

Linux

AIX

Sur les systèmes AIX and Linux , vous devez exécuter une commande **setmqinst** après l'installation pour définir l'installation comme installation principale.

Procédure



- Pour définir une installation comme installation principale, procédez comme suit:
 - a) Vérifiez si une installation est déjà l'installation principale en entrant la commande suivante:

```
MQ_INSTALLATION_PATH/bin/dspmqinst
```

où *MQ_INSTALLATION_PATH* est le chemin d'installation d'une installation IBM MQ .

- b) Si une installation IBM MQ existante est définie comme installation principale, [annulez-la](#) avant de passer à l'étape suivante.

c) Vérifiez que vous êtes connecté avec les droits appropriés:

-  En tant que superutilisateur sous AIX and Linux.
-  En tant que membre du groupe Administrateurs sur les systèmes Windows .

d) Entrez l'une des commandes suivantes :

- Pour définir l'installation principale à l'aide du chemin de l'installation que vous souhaitez utiliser comme installation principale:

```
MQ_INSTALLATION_PATH/bin/setmqinst -i -p MQ_INSTALLATION_PATH
```

- Pour définir l'installation principale à l'aide du nom de l'installation que vous souhaitez utiliser comme installation principale:

```
MQ_INSTALLATION_PATH/bin/setmqinst -i -n installationName
```

e) 

Sur les systèmes Windows , redémarrez le système.



- Pour annuler la définition d'une installation en tant qu'installation principale, procédez comme suit:

a) Vérifiez quelle installation est l'installation principale en entrant la commande suivante:

```
MQ_INSTALLATION_PATH/bin/dspmqinst
```

où *MQ_INSTALLATION_PATH* est le chemin d'installation d'une installation IBM MQ .

b) Vérifiez que vous êtes connecté avec les droits appropriés:

-  En tant que superutilisateur sous AIX and Linux.
-  En tant que membre du groupe Administrateurs sur les systèmes Windows .

- Entrez l'une des commandes suivantes :

- Pour annuler la définition de l'installation principale à l'aide du chemin de l'installation, vous ne souhaitez plus être l'installation principale:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -p MQ_INSTALLATION_PATH
```

- Pour annuler la définition de l'installation principale à l'aide du nom de l'installation, vous ne souhaitez plus être l'installation principale:

```
MQ_INSTALLATION_PATH/bin/setmqinst -x -n installationName
```

Tâches associées

[Désinstallation, mise à niveau et maintenance de l'installation principale](#)

[Choix du nom de l'installation](#)

Référence associée

[Fonctions disponibles uniquement avec l'installation principale sous Windows](#)

[Liens des commandes de contrôle et de la bibliothèque externe vers l'installation principale sous AIX and Linux](#)

[setmqinst](#)

Association d'un gestionnaire de files d'attente à une installation

Lorsque vous créez un gestionnaire de files d'attente, il est automatiquement associé à l'installation qui a émis la commande **crtmqm**. Sous AIX, Linux, and Windows, vous pouvez modifier l'installation associée à un gestionnaire de files d'attente à l'aide de la commande **setmqm**.

Pourquoi et quand exécuter cette tâche

L'installation à laquelle est associé un gestionnaire de files d'attente limite ce gestionnaire de façon à ce qu'il puisse être géré uniquement par les commandes de cette installation. Il existe trois exceptions clés:

- **setmqm** modifie l'installation associée au gestionnaire de files d'attente. Cette commande doit être émise à partir de l'installation que vous souhaitez associer au gestionnaire de files d'attente et non de l'installation à laquelle le gestionnaire de files d'attente est actuellement associé. Le nom d'installation spécifié par la commande **setmqm** doit correspondre à l'installation à partir de laquelle la commande est émise.
- **strmqm** doit être émis à partir de l'installation associée au gestionnaire de files d'attente.
- **dspmq** affiche des informations sur tous les gestionnaires de files d'attente d'un système, et pas seulement sur les gestionnaires de files d'attente associés à la même installation que la commande **dspmq**. La commande `dspmq -o installation` affiche des informations sur les gestionnaires de files d'attente associés à des installations.

Pour les environnements à haute disponibilité, la commande **addmqinf** associe automatiquement le gestionnaire de files d'attente à l'installation à partir de laquelle la commande **addmqinf** est émise. Tant que la commande **strmqm** est émise à partir de la même installation que la commande **addmqinf**, aucune configuration supplémentaire n'est requise. Pour démarrer le gestionnaire de files d'attente à l'aide d'une installation différente, vous devez d'abord modifier l'installation associée à l'aide de la commande **setmqm**.

Lorsque vous souhaitez associer un gestionnaire de files d'attente à une installation, vous pouvez utiliser la commande **setmqm** de l'une des manières suivantes:

- Déplacement de gestionnaires de files d'attente individuels entre des versions équivalentes de IBM MQ. Par exemple, le déplacement d'un gestionnaire de files d'attente d'un système de test vers un système de production.
- Migration de gestionnaires de files d'attente individuels depuis une ancienne version d' IBM MQ vers une version plus récente d' IBM MQ. La migration des gestionnaires de files d'attente entre les versions a diverses implications dont vous devez tenir compte. Pour plus d'informations sur la migration, voir [Gestion et migration](#).

Procédure

1. Arrêtez le gestionnaire de files d'attente à l'aide de la commande **endmqm** à partir de l'installation actuellement associée au gestionnaire de files d'attente.
2. Associez le gestionnaire de files d'attente à une autre installation à l'aide de la commande **setmqm** de cette installation.

Par exemple, pour définir le gestionnaire de files d'attente QMB à associer à une installation portant le nom `Installation2`, entrez la commande suivante à partir de `Installation2`:

```
MQ_INSTALLATION_PATH/bin/setmqm -m QMB -n Installation2
```

où `MQ_INSTALLATION_PATH` est le chemin dans lequel `Installation2` est installé.

3. Démarrez le gestionnaire de files d'attente à l'aide de la commande **strmqm** à partir de l'installation qui est maintenant associée au gestionnaire de files d'attente.

Cette commande effectue toute migration de gestionnaire de files d'attente nécessaire et permet au gestionnaire de files d'attente d'être prêt à être utilisé.

Que faire ensuite

Si l'installation à laquelle est associé un gestionnaire de files d'attente a été supprimée ou si les informations de statut du gestionnaire de files d'attente sont indisponibles, la commande **setmqm** ne parvient pas à associer le gestionnaire de files d'attente à une autre installation. Dans cette situation, effectuez les actions suivantes:

1. Utilisez la commande **dspmqinst** pour afficher les autres installations sur votre système.
2. Modifiez manuellement la zone `InstallationName` de la section `QueueManager` dans `mqs.ini` pour spécifier une autre installation.
3. Utilisez la commande **dlmqm** de cette installation pour supprimer le gestionnaire de files d'attente.

Concepts associés

«Recherche d'installations d'IBM MQ sur un système», à la page 503

Si vous disposez de plusieurs installations IBM MQ sur un système, vous pouvez vérifier quelles versions sont installées et où elles se trouvent.

«Fichier de configuration IBM MQ , `mqs.ini`», à la page 98

Le fichier de configuration IBM MQ , `mqs.ini`, contient des informations relatives à tous les gestionnaires de files d'attente sur le noeud. Il est créé automatiquement lors de l'installation.

Tâches associées

[Choix d'une installation principale](#)

Référence associée

[addmqinf](#)

[dspmq](#)

[dspmqinst](#)

[endmqm](#)

[setmqm](#)

[strmqm](#)

ALW

Recherche d'installations d'IBM MQ sur un système

Si vous disposez de plusieurs installations IBM MQ sur un système, vous pouvez vérifier quelles versions sont installées et où elles se trouvent.

Vous pouvez utiliser les méthodes suivantes pour rechercher les installations IBM MQ sur votre système:

- Utilisez les outils d'installation de plateforme pour déterminer où IBM MQ a été installé. Utilisez ensuite la commande **dspmqrver** à partir d'une installation IBM MQ . Les commandes suivantes sont des exemples de commandes que vous pouvez utiliser pour déterminer où IBM MQ a été installé:

- **AIX** Sur les systèmes AIX , vous pouvez utiliser la commande **lslpp** :

```
lslpp -R ALL -l mqm.base.runtime
```

- **Linux** Sur les systèmes Linux , vous pouvez utiliser la commande **rpm** :

```
rpm -qa --qf "%{NAME}-%{VERSION}-%{RELEASE}\t%{INSTPREFIXES}\n" | grep MQSeriesRuntime
```

- **Windows** Sur les systèmes Windows , vous pouvez utiliser la commande **wmic** . Cette commande peut installer le client wmic:

```
wmic product where "(Name like '%MQ%') AND (not Name like '%bitSupport')" get Name, Version, InstallLocation
```

- **Linux** **AIX** Sur les systèmes AIX and Linux , exécutez la commande suivante pour savoir où IBM MQ a été installé:

```
cat /etc/opt/mqm/mqinst.ini
```

Utilisez ensuite la commande **dspmqr** à partir d'une installation IBM MQ .

- **Windows** Pour afficher les détails des installations sur le système, sous Windows32 bits, exécutez la commande suivante:

```
reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\IBM\WebSphere MQ\Installation" /s
```

- **Windows** Sous Windows64 bits, exécutez la commande suivante:

```
reg.exe query "HKEY_LOCAL_MACHINE\SOFTWARE\Wow6432Node\IBM\WebSphere MQ\Installation" /s
```

Référence associée

[dspmqr](#)

[dspmqrinst](#)

[Installations multiples](#)

Configuration de la haute disponibilité, de la reprise et du redémarrage

Vous pouvez rendre vos applications hautement disponibles en gérant la disponibilité des files d'attente en cas de défaillance d'un gestionnaire de files d'attente et en récupérant les messages après une défaillance du serveur ou de l'espace de stockage.

Pourquoi et quand exécuter cette tâche

• **z/OS** Sous z/OS, la haute disponibilité est intégrée à la plateforme. Voir [Files d'attente partagées et groupes de partage de files d'attente](#).

• **Multi** Sous [Multiplateformes](#), vous pouvez améliorer la disponibilité de l'application client en utilisant la reconnexion du client pour basculer automatiquement un client entre un groupe de gestionnaires de files d'attente ou vers la nouvelle instance active d'un gestionnaire de files d'attente multi-instance après une défaillance du gestionnaire de files d'attente. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java. Un gestionnaire de files d'attente multi-instance est configuré pour s'exécuter en tant que gestionnaire de files d'attente unique sur plusieurs serveurs. Vous déployez des applications serveur sur ce gestionnaire de files d'attente. Si le serveur exécutant l'instance active échoue, l'exécution est automatiquement basculée vers une instance de secours du même gestionnaire de files d'attente sur un autre serveur. Si vous configurez des applications serveur pour qu'elles s'exécutent en tant que services de gestionnaire de files d'attente, elles sont redémarrées lorsqu'une instance de secours devient l'instance de gestionnaire de files d'attente active.

Une autre façon d'augmenter la disponibilité des applications serveur sur Multiplatforms consiste à déployer des applications serveur sur plusieurs ordinateurs d'un cluster de gestionnaires de files d'attente. A partir de la IBM WebSphere MQ 7.1 , la reprise sur incident de cluster réexécute les opérations qui ont généré des problèmes jusqu'à ce que les problèmes soient résolus. Voir [«Modifications apportées à la reprise sur incident de cluster sur les serveurs sur Multiplatforms»](#), à la page 745. Vous pouvez également configurer IBM MQ for Multiplatforms dans le cadre d'une solution de mise en cluster spécifique à la plateforme, telle que:

- Microsoft Cluster Server
- **IBM i** Clusters à haute disponibilité sur IBM i

- **Linux** **AIX** PowerHA for AIX (anciennement HACMP on AIX) et autres solutions de mise en cluster UNIX and Linux

Linux Sur les systèmes Linux , vous pouvez configurer des gestionnaires de files d'attente de données répliquées (RDQM) pour implémenter des solutions de haute disponibilité ou de reprise après incident. Pour la haute disponibilité, les instances du même gestionnaire de files d'attente sont configurées sur chaque noeud d'un groupe de trois serveurs Linux . L'une des trois instances est l'instance active. Les données du gestionnaire de files d'attente actif sont répliquées de manière synchrone sur les deux autres instances, de sorte que l'une de ces instances peut prendre le relais en cas de défaillance. Pour la reprise après incident, un gestionnaire de files d'attente s'exécute sur un noeud principal sur un site, avec une instance secondaire de ce gestionnaire de files d'attente située sur un noeud de reprise sur un autre site. Les données sont répliquées entre l'instance principale et l'instance secondaire, et si le noeud principal est perdu pour une raison quelconque, l'instance secondaire peut être transformée en instance principale et démarrée.

CP4I Native HA est une solution à haute disponibilité destinée aux conteneurs. Native HA utilise la réplication de journal pour maintenir à jour trois instances d'un gestionnaire de files d'attente s'exécutant sur des noeuds différents. Une instance est active à la fois et traite les messages. Le gestionnaire de files d'attente actif envoie ses mises à jour de journal aux deux autres instances pour les maintenir à jour. Si l'instance active échoue, l'une des instances de réplique prend automatiquement le rôle actif.

MQ Appliance Une autre option pour une solution de haute disponibilité ou de reprise après incident consiste à déployer une paire de dispositifs IBM MQ . Voir [High Availability](#) et [Disaster Recovery](#) dans la documentation IBM MQ Appliance .

Un système de messagerie garantit que les messages entrés dans le système sont distribués à leur destination. IBM MQ peut tracer la route d'un message lorsqu'il passe d'un gestionnaire de files d'attente à un autre à l'aide de la commande **dspmqzte** . En cas de défaillance d'un système, les messages peuvent être récupérés de différentes manières en fonction du type de défaillance et de la manière dont un système est configuré. IBM MQ gère les journaux de reprise des activités des gestionnaires de files d'attente qui gèrent la réception, la transmission et la distribution des messages. Il utilise ces journaux pour trois types de reprise:

1. *Redémarrez la reprise*, lorsque vous arrêtez IBM MQ de manière planifiée.
2. *Reprise après incident*, lorsqu'un incident s'arrête IBM MQ.
3. *Reprise sur incident lié au support*, pour restaurer les objets endommagés.

Dans tous les cas, la récupération restaure le gestionnaire de files d'attente à l'état dans lequel il se trouvait lorsque le gestionnaire de files d'attente s'est arrêté, sauf que les transactions en cours sont annulées et que les mises à jour qui étaient en cours au moment où le gestionnaire de files d'attente s'est arrêté sont supprimées des files d'attente. La reprise restaure tous les messages persistants ; les messages non persistants peuvent être perdus pendant le processus.



ATTENTION : Vous ne pouvez pas déplacer les journaux de reprise vers un autre système d'exploitation.

reconnexion client automatique

Vous pouvez faire en sorte que vos applications client se reconnectent automatiquement, sans écrire de code supplémentaire, en configurant un certain nombre de composants.

La reconnexion automatique du client est *intégrée*. La connexion est restaurée automatiquement à tout moment dans le programme d'application client et les descripteurs permettant d'ouvrir les objets sont tous restaurés.

Par contre, la reconnexion manuelle requiert que l'application client recrée une connexion avec MQCONN ou MQCONNX et qu'elle rouvre les objets. La reconnexion automatique du client est adaptée pour de nombreuses applications client, mais pas toutes.

Le [Tableau 31](#), à la [page 506](#) répertorie l'édition la plus ancienne de la prise en charge du client IBM MQ qui doit être installée sur un poste de travail client. Vous devez mettre à niveau les postes de travail client vers l'un de ces niveaux pour qu'une application utilise la reconnexion client automatique. Le [Tableau 32](#), à la [page 507](#) répertorie les autres conditions requises pour activer la reconnexion automatique du client.

Avec l'accès du programme aux options de reconnexion, une application client peut définir des options de reconnexion. A l'exception des clients JMS et XMS , si une application client a accès aux options de reconnexion, elle peut également créer un gestionnaire d'événements pour gérer les événements de reconnexion.

Une application client existante peut bénéficier de la prise en charge de la reconnexion, sans recompilation ni liaison:

- Pour un client nonJMS , définissez la variable d'environnement `mqclient.ini DefRecon` pour définir les options de reconnexion. Utilisez une table de définition de canal du client pour vous connecter à un gestionnaire de files d'attente. Si le client doit se connecter à un gestionnaire de files d'attente multi-instance, indiquez les adresses réseau des instances de gestionnaire de files d'attente actives et de secours dans la table de définition de canal du client. Pour un gestionnaire de files d'attente de données répliquées ou un gestionnaire de files d'attente à haute disponibilité sur un dispositif IBM MQ , vous pouvez spécifier une adresse IP flottante utilisée par les gestionnaires de files d'attente actifs et de secours pour simplifier la configuration.
- Pour un client JMS , définissez les options de reconnexion dans la configuration de la fabrique de connexions. Lors de l'exécution dans le conteneur d'EJB d'un serveur Java EE , les beans gérés par message peuvent se reconnecter à IBM MQ à l'aide du mécanisme de reconnexion fourni par les spécifications d'activation de l'adaptateur de ressources IBM MQ (ou des ports d'écoute s'ils s'exécutent dans WebSphere Application Server). Toutefois, si l'application n'est pas un bean géré par message (ou qu'elle est en cours d'exécution dans le conteneur Web), elle doit implémenter sa propre logique de reconnexion car la reconnexion automatique du client n'est pas prise en charge dans ce scénario. L'adaptateur de ressources IBM MQ offre cette capacité de reconnexion pour la distribution des messages aux beans gérés par message, mais d'autres éléments Java EE , tels que les servlets, doivent implémenter leur propre reconnexion.

Remarque : La reconnexion automatique du client n'est pas prise en charge par IBM MQ classes for Java.

Interface client	Environnement	Accès au programme pour les options de reconnexion	Prise en charge de la reconnexion
API de messagerie	C, C ++, COBOL, Unmanaged Visual Basic, XMS (XMS non géré sous Windows)	7.0.1	7.0.1
	JMS (conteneur client JSE et Java EE et conteneurs gérés)	7.0.1.3	7.0.1.3
	IBM MQ classes for Java	Non pris en charge	Non pris en charge
	Clients XMS gérés et .NET gérés: C#, Visual Basic,	7.1	7.1

Tableau 31. Clients pris en charge (suite)

Interface client	Environnement	Accès au programme pour les options de reconnexion	Prise en charge de la reconnexion
Autres API	Windows Communication Foundation (non géré ¹)	Non pris en charge	7.0.1
	Windows Communication Foundation (géré ¹)	Non pris en charge	Non pris en charge
	Axe 1	Non pris en charge	Non pris en charge
	Axe 2	Non pris en charge	7.0.1.3
	HTTP (Web 2.0)	Non pris en charge	7.0.1.3

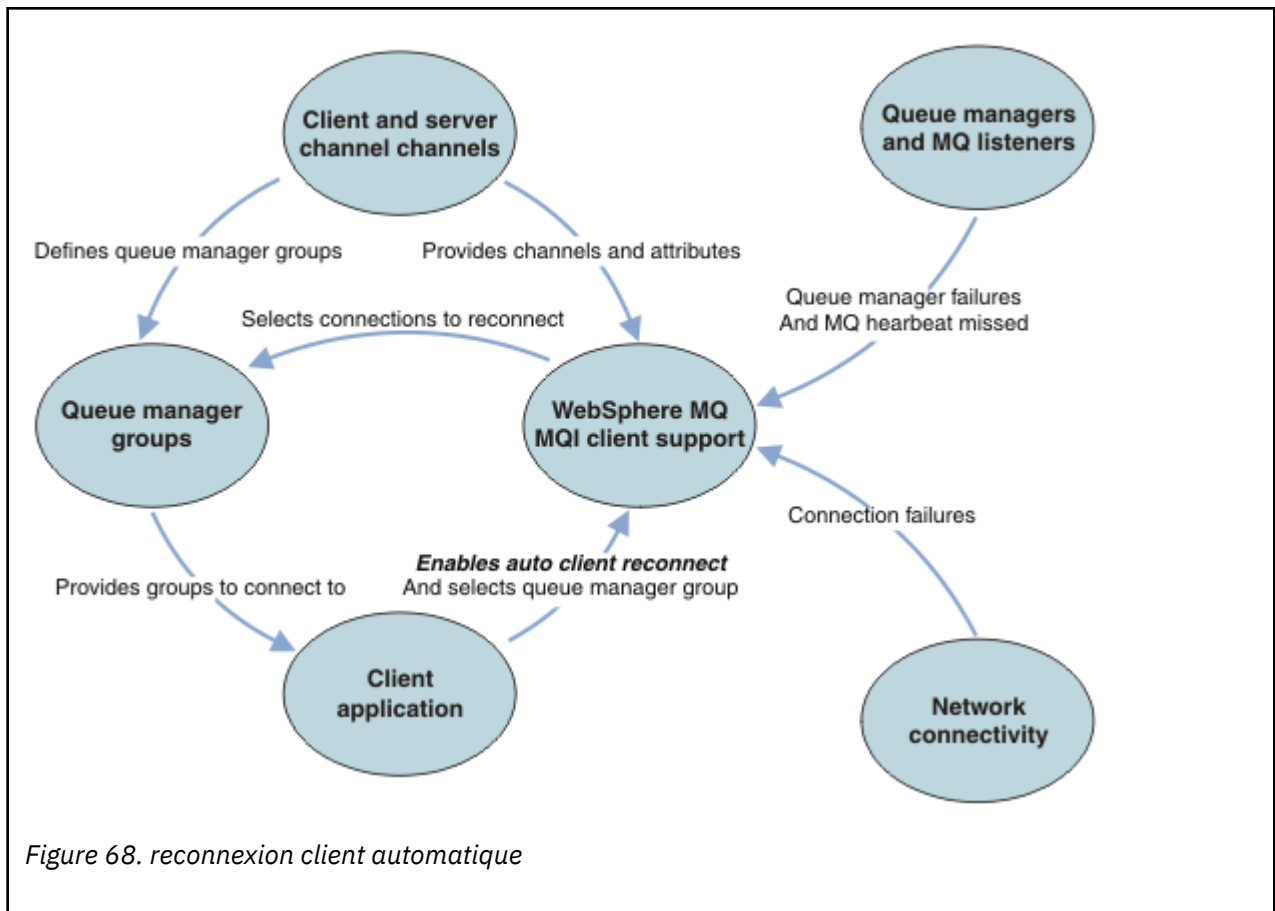
1. Définissez le mode géré ou non géré dans la configuration de liaison WCF.

La reconnexion automatique a les exigences de configuration suivantes:

Tableau 32. Configuration requise pour la reconnexion automatique

Composant	Exigence	Effet du non-respect de l'exigence
Installation sur IBM MQ MQI client	Voir Tableau 31 , à la page 506	MQRC_OPTIONS_ERROR
IBM MQ Installation du serveur	Niveau 7.0.1	MQRC_OPTIONS_ERROR
Canal	SHARECNV > 0	MQRC_ENVIRONMENT_ERROR
environnement d'application	Doit être une unité d'exécution	MQRC_ENVIRONMENT_ERROR
Interface MQI	L'une des entrées suivantes : <ul style="list-style-type: none"> MQCONNX avec MQCNO Options défini sur MQCNO_RECONNECT ou MQCNO_RECONNECT_Q_MGR. Defrecon=YES QMGR dans mqclient.ini Dans JMS , définissez la propriété CLIENTRECONNECTOPTIONS de la fabrique de connexions. 	MQCC_FAILED lorsqu'une connexion est interrompue ou qu'un gestionnaire de files d'attente s'arrête ou échoue.

Le [Figure 68](#), à la page 508 présente les principales interactions entre les composants impliqués dans la reconnexion du client.



Application du client

L'application client est un IBM MQ MQI client. Pour plus de détails sur la reconnexion automatique du client pour un client JMS, voir [Utilisation de la reconnexion automatique du client JMS](#).

- Par défaut, les clients ne sont pas automatiquement reconnectés. Activez la reconnexion automatique du client en définissant l'option MQCONNX MQCNO MQCNO_RECONNECT ou MQCNO_RECONNECT_Q_MGR.
- De nombreuses applications sont écrites de manière à pouvoir tirer parti de la reconnexion automatique sans codage supplémentaire. Activez la reconnexion automatique pour les programmes existants, sans apporter de modifications au codage, en définissant l'attribut DefRecon dans la strophe channels du fichier de configuration mqclient.ini.
- Utilisez l'une des trois options suivantes:
 1. Modifiez le programme de sorte que la logique ne soit pas affectée par la reconnexion. Par exemple, vous pouvez être amené à émettre des appels MQI dans le point de synchronisation, puis à soumettre à nouveau les transactions annulées. Les consommateurs asynchrones doivent vérifier s'ils ont été "suspendus" si une transaction est annulée.
 2. Ajoutez un gestionnaire d'événements pour détecter la reconnexion et restaurez l'état de l'application client lorsque la connexion est rétablie.
 3. N'activez pas la reconnexion automatique: déconnectez le client et émettez un nouvel appel MQCONN ou MQCONNX MQI pour trouver une autre instance de gestionnaire de files d'attente qui s'exécute dans le même groupe de gestionnaires de files d'attente.

Pour plus de détails sur ces trois options, voir «[Récupération d'application](#)», à la page 609.

- La reconnexion à un gestionnaire de files d'attente du même nom ne garantit pas que vous vous êtes reconnecté à la même instance d'un gestionnaire de files d'attente.

Utilisez l'option MQCNO MQCNO_RECONNECT_Q_MGR pour vous reconnecter à une instance du même gestionnaire de files d'attente.

- Un client peut enregistrer un gestionnaire d'événements afin qu'il puisse être informé de l'état de la reconnexion. Le MQHCONN transmis dans le gestionnaire d'événements ne peut pas être utilisé. Les codes anomalie suivants sont fournis:

MQRC_RECONNEXION

La connexion a échoué et le système tente de se reconnecter. Vous recevez plusieurs événements MQRC_RECONNECTING si plusieurs tentatives de reconnexion sont effectuées.

MQRC_RECONNECTE

La reconnexion effectuée et tous les descripteurs ont été rétablies.

Echec de MQRC_RECONNECT_FAILED

La reconnexion a échoué.

MQRC_RECONNECT_QMID_DISCORDANCE

Une connexion reconnectable a spécifié MQCNO_RECONNECT_Q_MGR et la connexion a tenté de se reconnecter à un autre gestionnaire de files d'attente.

MQRC_RECONNECT_Q_MGR_REQD

Une option, telle que MQMO_MATCH_MSG_TOKEN dans un appel MQGET , a été spécifiée dans le programme client qui nécessite une reconnexion au même gestionnaire de files d'attente.

- Un client reconnectable ne peut se reconnecter automatiquement qu'après la connexion. C'est-à-dire que l'appel MQCONNX lui-même n'est pas relancé en cas d'échec. Par exemple, si vous recevez le code retour 2543 - MQRC_STANDBY_Q_MGR de MQCONNX, émettez à nouveau l'appel après un bref délai.

MQRC_RECONNECT_INCOMPATIBLE

Ce code anomalie est renvoyé lorsque l'application tente d'utiliser MQPMO_LOGICAL_ORDER (avec MQPUT et MQPUT1) ou MQGMO_LOGICAL_ORDER (avec MQGET) lorsque les options de reconnexion sont définies. La raison pour laquelle le code anomalie est renvoyé est de s'assurer que les applications n'utilisent jamais la reconnexion dans de tels cas.

MQRC_CALL_INTERROMPUE

Ce code anomalie est renvoyé lorsque la connexion est interrompue lors de l'exécution de l'appel de validation et que le client se reconnecte. Une instruction MQPUT d'un message persistant en dehors du point de synchronisation entraîne également le renvoi du même code anomalie à l'application.

Gestionnaires de files d'attente à haute disponibilité

Les gestionnaires de files d'attente à haute disponibilité possèdent une instance active et une ou plusieurs instances de secours d'un gestionnaire de files d'attente. Le gestionnaire de files d'attente actif est synchronisé avec les gestionnaires de files d'attente de secours, de sorte qu'un gestionnaire de files d'attente de secours peut automatiquement prendre le relais en cas d'échec de l'instance active. Il existe un certain nombre de solutions différentes pour fournir des gestionnaires de files d'attente à haute disponibilité. Voir [«Configurations à haute disponibilité»](#), à la page 516.

Vous pouvez simplifier le redémarrage des applications IBM MQ MQI client , une fois qu'un gestionnaire de files d'attente à haute disponibilité a activé son instance de secours, à l'aide de la reconnexion automatique du client.

L'instance de secours d'un gestionnaire de files d'attente à haute disponibilité se trouve généralement à une adresse réseau différente de celle de l'instance active. Incluez les adresses réseau des deux instances dans la table de définition de connexion du client (CCDT). Indiquez une liste d'adresses réseau pour le paramètre **CONNNAME** ou définissez plusieurs lignes pour le gestionnaire de files d'attente dans la table de définition de canal du client. Les gestionnaires de files d'attente de données répliquées et les gestionnaires de files d'attente à haute disponibilité du dispositif IBM MQ prennent en charge les adresses IP flottantes, où vous spécifiez une adresse unique à utiliser avec les gestionnaires de files d'attente actifs ou de secours.

Groupes de gestionnaires de files d'attente

Généralement, IBM MQ MQI clients se reconnecte à n'importe quel gestionnaire de files d'attente d'un groupe de gestionnaires de files d'attente. Parfois, vous souhaitez qu'un IBM MQ MQI client se reconnecte


uniquement au même gestionnaire de files d'attente. Il peut avoir une affinité avec un gestionnaire de files d'attente.

Vous pouvez indiquer si l'application client se connecte et se reconnecte toujours à un gestionnaire de files d'attente du même nom, au même gestionnaire de files d'attente ou à l'un des ensembles de gestionnaires de files d'attente définis avec la même valeur QMNAME dans la table des connexions client.

- L'attribut de nom de gestionnaire de files d'attente, QMNAME, dans la définition de canal du client est le nom d'un groupe de gestionnaires de files d'attente.
- Dans votre application client, si vous définissez la valeur du paramètre MQCONN ou MQCONNX QmgrName sur un nom de gestionnaire de files d'attente, le client se connecte uniquement aux gestionnaires de files d'attente portant ce nom. Si vous préfixez le nom du gestionnaire de files d'attente avec un astérisque (*), le client se connecte à n'importe quel gestionnaire de files d'attente du groupe de gestionnaires de files d'attente avec la même valeur QMNAME . Pour une explication complète, voir [Groupes de gestionnaires de files d'attente dans la table de définition de canal du client](#).

Vous pouvez empêcher un client de se reconnecter à un autre gestionnaire de files d'attente. Définissez l'option MQCNO , MQCNO_RECONNECT_Q_MGR. Le IBM MQ MQI client échoue s'il se reconnecte à un autre gestionnaire de files d'attente. Si vous définissez l'option MQCNO , MQCNO_RECONNECT_Q_MGR, n'incluez pas d'autres gestionnaires de files d'attente dans le même groupe de gestionnaires de files d'attente. Le client renvoie une erreur si le gestionnaire de files d'attente auquel il se reconnecte n'est pas le même que celui auquel il s'est connecté.

Groupes de partage de files d'attente

 La reconnexion automatique du client aux groupes de partage de files d'attente z/OS utilise les mêmes mécanismes de reconnexion que tout autre environnement. Le client se reconnecte à la même sélection de gestionnaires de files d'attente que celle configurée pour la connexion d'origine. Par exemple, lors de l'utilisation de la table de définition de canal du client, l'administrateur doit s'assurer que toutes les entrées de la table sont résolues dans le même groupe de partage de files d'attente z/OS .

Définitions de canal client et serveur

Les définitions de canal client et serveur définissent les groupes de gestionnaires de files d'attente auxquels une application client peut se reconnecter. Les définitions régissent la sélection et la temporisation des reconnexions, ainsi que d'autres facteurs, tels que la sécurité ; voir les rubriques connexes. Les attributs de canal les plus pertinents à prendre en compte pour la reconnexion sont répertoriés dans deux groupes:

Attributs de connexion client

Affinité de connexion (AFFINITY) AFFINITY

Affinité de connexion

Poids du canal client (CLNTWGHT) CLNTWGHT

Poids du canal client.

Nom de connexion (CONNAME) CONNAME

Informations de connexion.

Intervalle des pulsations (HBINT) HBINT

Intervalle des pulsations. Définissez l'intervalle des pulsations sur le canal de connexion serveur.

Keepalive Interval (KAINT) KAINT

Intervalle de signal de présence. Définissez l'intervalle de signal de présence sur le canal de connexion serveur.

 Notez que KAINT s'applique à z/OS uniquement.

Nom du gestionnaire de files d'attente (QMNAME) QMNAME

Nom du gestionnaire de files d'attente

Attributs de connexion au serveur

Intervalle des pulsations (HBINT) HBINT

Intervalle des pulsations. Définissez l'intervalle des pulsations sur le canal de connexion client.

Keepalive Interval (KAIN) KAIN

Intervalle de signal de présence. Définissez l'intervalle de signal de présence sur le canal de connexion client.

 Notez que KAIN s'applique à z/OS uniquement.

KAIN est un signal de présence de couche réseau et HBINT est un signal de présence IBM MQ entre le client et le gestionnaire de files d'attente. La définition de ces pulsations sur une durée plus courte a deux objectifs:

1. En simulant l'activité sur la connexion, le logiciel de couche réseau responsable de la fermeture des connexions inactives est moins susceptible d'arrêter votre connexion.
2. Si la connexion est arrêtée, le délai avant la détection de l'interruption de la connexion est raccourci.

L'intervalle de signal de présence TCP/IP par défaut est de deux heures. Envisagez de définir les attributs KAIN et HBINT sur une durée plus courte. Ne supposez pas que le comportement normal d'un réseau répond aux besoins d'une reconnexion automatique. Par exemple, certains pare-feux peuvent arrêter une connexion TCP/IP inactive après seulement 10 minutes.

Connectivité réseau

Seuls les incidents réseau transmis à IBM MQ MQI client par le réseau sont traités par la fonction de reconnexion automatique du client.

- Les reconnexions effectuées automatiquement par le transport sont invisibles pour IBM MQ.
- La définition de HBINT permet de gérer les incidents réseau invisibles pour IBM MQ.

Gestionnaires de files d'attente et programmes d'écoute IBM MQ

La reconnexion du client est déclenchée par une défaillance du serveur, une défaillance du gestionnaire de files d'attente, une défaillance de la connectivité du réseau et par le basculement d'un administrateur vers une autre instance du gestionnaire de files d'attente.

- Si vous utilisez un gestionnaire de files d'attente multi-instance, une cause supplémentaire de reconnexion du client se produit lorsque vous passez de l'instance de gestionnaire de files d'attente active à une instance de secours.
- L'arrêt d'un gestionnaire de files d'attente à l'aide de la commande **endmqm** par défaut ne déclenche pas la reconnexion automatique du client. Ajoutez l'option **-r** à la commande **endmqm** pour demander la reconnexion automatique du client ou l'option **-s** pour effectuer le transfert vers une instance de gestionnaire de files d'attente de secours après l'arrêt.

Prise en charge de la reconnexion automatique IBM MQ MQI client

Si vous utilisez la prise en charge de la reconnexion automatique du client dans IBM MQ MQI client, l'application client se reconnecte automatiquement et poursuit le traitement sans que vous ayez à émettre un appel MQCONN ou MQCONNX MQI pour vous reconnecter au gestionnaire de files d'attente.

- La reconnexion automatique du client est déclenchée par l'une des occurrences suivantes:
 - Echec du gestionnaire de files d'attente
 - Arrêt d'un gestionnaire de files d'attente et spécification de l'option **-r**, **reconnect**, dans la commande **endmqm**
- Les options MQCONNX MQCNO contrôlent si vous avez activé la reconnexion automatique du client. Les options sont décrites dans [Options de reconnexion](#).
- La reconnexion automatique du client émet des appels MQI pour le compte de votre application afin de restaurer le descripteur de connexion et les descripteurs vers d'autres objets ouverts, de sorte que votre programme puisse reprendre le traitement normal après avoir traité les erreurs MQI résultant de l'interruption de la connexion. Voir [«Récupération d'un client reconnecté automatiquement»](#), à la page [611](#).

- Si vous avez écrit un programme d'exit de canal pour la connexion, l'exit reçoit ces appels MQI supplémentaires.
- Vous pouvez enregistrer un gestionnaire d'événements de reconnexion, qui est déclenché lorsque la reconnexion commence et lorsqu'elle se termine.

Bien que le temps de reconnexion prévu ne soit pas supérieur à une minute, la reconnexion peut prendre plus de temps car un gestionnaire de files d'attente peut avoir de nombreuses ressources à gérer. Pendant ce temps, une application client peut détenir des verrous qui n'appartiennent pas à des ressources IBM MQ . Il existe une valeur de délai d'attente que vous pouvez configurer pour limiter le délai d'attente d'un client pour la reconnexion. La valeur (en secondes) est définie dans le fichier `mqclient.ini`.

```
Channels:
MQReconnectTimeout = 1800
```

Aucune tentative de reconnexion n'est effectuée après l'expiration du délai d'attente. Lorsque le système détecte que le délai d'attente a expiré, il renvoie une erreur `MQRC_RECONNECT_FAILED`.

Concepts associés

[Clients reconnectables](#)

Tâches associées

[Arrêt d'un gestionnaire de files d'attente](#)

z/OS

Console message monitoring

On IBM MQ for z/OS, there are a number of information messages issued by the queue manager or channel initiator that should be considered particularly significant. These messages do not in themselves indicate a problem, but can be useful in tracking because they do indicate a potential issue which might need addressing.

The presence of these console messages might also indicate that a user application is putting a large number of messages to the page set, which might be a symptom of a larger problem:

- A problem with the user application which PUTs messages, such as an uncontrolled loop.
- A user application which GETs the messages from the queue is no longer functioning.

Console messages to monitor

The following list outlines messages which can potentially indicate larger problems. Determine if it is necessary to track these messages with system automation and provide appropriate documentation so any potential problems can be followed up effectively.

CSQI004I: csect-name CONSIDER INDEXING queue-name BY index-type FOR connection-type CONNECTION connection-name, num-msgs MESSAGES SKIPPED

- The queue manager has detected an application receiving messages by message ID or correlation ID from a queue that does not have an index defined.
- Consider establishing an index for the identified queue by altering the local queue object, *queue-name*, `INDXTYPE` attribute to have value *index-type*.

CSQI031I: csect-name THE NEW EXTENT OF PAGE SET psid HAS FORMATTED SUCCESSFULLY

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of the failure to process the messages.

CSQI041I: csect-name JOB jobname USER userid HAD ERROR ACCESSING PAGE SET psid

- Determine if the page set is allocated to the queue manager.
- Issue a **DISPLAY USAGE** command to determine the state of the page set.

- Check the queue manager joblog for additional error messages.

CSQI045I: csect-name Log RBA has reached rba. Plan a log reset

- Plan to stop the queue manager at a convenient time and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQI046E: csect-name Log RBA has reached rba. Perform a log reset

- Plan to stop the queue manager at a convenient time and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQI047E: csect-name Log RBA has reached rba. Stop queue manager and reset logs

- Stop the queue manager immediately and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQJ004I: ACTIVE LOG COPY *n* INACTIVE, LOG IN SINGLE MODE, ENDRBA= *ttt*

- The queue manager has activated 'single' logging mode. This is often indicative of a log offload problem.
- Issue a **DISPLAY LOG** command to determine your settings for duplexing of active and archive logs. This display also shows how many active logs need offload processing.
- Check the queue manager joblog for additional error messages

CSQJ031D: csect-name, THE LOG RBA RANGE MUST BE RESET. REPLY 'Y' TO CONTINUE STARTUP OR 'N' TO SHUTDOWN

- Stop the queue manager and reset the logs as soon as possible and reset the logs.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQJ032E: csect-name alert-lvl - APPROACHING END OF THE LOG RBA RANGE OF *max-rba*. CURRENT LOG RBA IS *current-rba*.

- Plan to stop the queue manager and reset the logs as soon as possible.
- If your queue manager is using 6-byte log RBAs, consider converting the queue manager to use 8-byte log RBAs.

CSQJ110E: LAST COPY_{*n*} ACTIVE LOG DATA SET IS *nnn* PERCENT FULL

- Take steps to complete other waiting offload tasks by performing a display request to determine the outstanding requests related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue.
- Consider whether there are sufficient active log data sets. If necessary, you can add additional log data sets dynamically by using the DEFINE LOG command.

CSQJ111A: OUT OF SPACE IN ACTIVE LOG DATA SETS

- Perform a display request to ensure that there are no outstanding requests that are related to the log offload process. Take the necessary action to satisfy any requests, and permit offload to continue.
- Consider whether there are sufficient active log data sets. If necessary, you can add additional log data sets dynamically by using the DEFINE LOG command.
- If the delay was caused by the lack of a resource required for offload, the necessary resource must be made available to allow offload to complete and thus permit logging to proceed. For information about recovery from this condition, see Archive log problems.

CSQJ114I: ERROR ON ARCHIVE DATA SET, OFFLOAD CONTINUING WITH ONLY ONE ARCHIVE DATA SET BEING GENERATED

- Check the queue manager joblog for additional error messages.
- Make a second copy of the archive log and update your BSDS manually.

CSQJ115E: OFFLOAD FAILED, COULD NOT ALLOCATE AN ARCHIVE DATA SET

Review the error status information of message CSQJ103E or CSQJ073E. Correct the condition that caused the data set allocation error so that, on retry, the offload can take place.

CSQJ136I: UNABLE TO ALLOCATE TAPE UNIT FOR CONNECTION-ID= *xxxx* CORRELATION-ID= *yyyyyy*, *m* ALLOCATED *n* ALLOWED

- Check the queue manager joblog for additional error messages.

CSQJ151I: *csect-name* ERROR READING RBA *rrr*, CONNECTION-ID= *xxxx* CORRELATION-ID= *yyyyyy* REASON CODE= *ccc*

- Check the queue manager joblog for additional messages.
- Issue a **DISPLAY CONN** command to determine which connection is not committing its activity.
- Ensure the application can commit its updates.

CSQJ160I: LONG-RUNNING UOW FOUND, URID= *urid* CONNECTION NAME= *name*

- Check the queue manager joblog for additional messages.
- Issue a **DISPLAY CONN** command to determine which connection is not committing its activity.
- Ensure the application can commit its updates.

CSQJ161I: UOW UNRESOLVED AFTER *n* OFFLOADS, URID= *urid* CONNECTION NAME= *name*

- Determine if the page set is allocated to the queue manager.
- Issue a **DISPLAY USAGE** command to determine the state of the page set.
- Check the queue manager joblog for additional messages.

CSQP011E: CONNECT ERROR STATUS *ret-code* FOR PAGE SET *psid*

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of the failure to process messages.

CSQP013I: *csect-name* NEW EXTENT CREATED FOR PAGE SET *psid*. NEW EXTENT WILL NOW BE FORMATTED

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of failure to process messages.
- Determine if queues need to be relocated to another page set.
- If the volume is full, determine if you need to make the page set a multi volume data set. If the page set is already multi-volume, consider adding more volumes to the storage group being used. Once more space is available retry the expansion by setting the page set **EXPAND** method to **SYSTEM**. If a retry is required, toggle **EXPAND** to **SYSTEM** and then back to your normal setting.

CSQP014E: *csect-name* EXPANSION FAILED FOR PAGE SET *psid*. FUTURE REQUESTS TO EXTEND IT WILL BE REJECTED

- Check the curdepth of the queues allocated to this page set.
- Investigate the cause of failure to process messages.
- Determine if queues need to be relocated to another page set.

CSQP016E: *csect-name* PAGE SET *psid* HAS REACHED THE MAXIMUM NUMBER OF EXTENTS. IT CANNOT BE EXTENDED AGAIN

- Check the curdepth of the queues allocated to this page set.

- Investigate the cause of failure to process messages.

CSQP017I: csect-name EXPANSION STARTED FOR PAGE SET *psid*

Issue DISPLAY THREAD commands to determine the state of the Units of Work in IBM MQ.

CSQP047E: Unavailable page sets can cause problems - take action to correct this situation

- Follow the system programmer response.

CSQQ008I: nn units of recovery are still in doubt in queue manager *qqqq*

- Investigate the state of your dead letter queue. Ensure the dead letter queue is not PUT disabled.
- Ensure the dead letter queue is not at the MAXMSG limit.

CSQQ113I: *psb-name region-id* This message cannot be processed

- Check the CSQOUTX data set to determine the cause of the CSQINPX failure.
- Some commands may not be processed.

CSQX035I: csect-name Connection to queue manager *qmgr-name* stopping or broken, MQCC= *mqcc* MQRC= *mqrc* (*mqrc-text*)

- Check the MQRC to determine the cause of the failure.
- These codes are documented in [IBM MQ for z/OS messages, completion, and reason codes](#).

CSQX032I: csect-name Initialization command handler terminated

- Check the MQRC to determine the cause of the failure.
- These codes are documented in [IBM MQ for z/OS messages, completion, and reason codes](#).

CSQX048I: csect-name Unable to convert message for *name*, MQCC= *mqcc* MQRC= *mqrc* (*mqrc-text*)

- Check the joblog to determine the cause of the TCP/IP failure.
- Check the TCP/IP address space for errors.

CSQX234I: csect-name Listener stopped, TRPTYPE= *trptype* INDISP= *disposition*

- If the listener does not stop, following a **STOP** command, check the TCP/IP address space for errors.
- Follow the system programmer response.

CSQX407I: csect-name Cluster queue *q-name* definitions inconsistent

- Multiple cluster queues within the cluster have inconsistent values. Investigate and resolve the differences.

CSQX411I: csect-name Repository manager stopped

- If the repository manager has stopped because of an error, check the joblog for messages.

CSQX417I: csect-name Cluster-senders remain for removed queue manager *qmgr-name*

- Follow the system programmer response.

CSQX418I: csect-name Only one repository for cluster *cluster_name*

- For increased high availability, clusters should be configured with two full repositories.

CSQX419I: csect-name No cluster-receivers for cluster *cluster_name*

- Follow the system programmer response.

CSQX420I: csect-name No repositories for cluster *cluster_name*

- Follow the system programmer response.

CSQX448E: csect-name Repository manager stopping because of errors. Restart in *n* seconds

- Follow the system programmer response.

This message is put out every 600 seconds (10 minutes) until the SYSTEM.CLUSTER.COMMAND.QUEUE is enabled, by using the command:

```
ALTER QLOCAL (SYSTEM.CLUSTER.COMMAND.QUEUE) GET (ENABLED)
```

Before enabling the queue, manual intervention might be required to resolve the problem that caused the repository manager to end, prior to the first CSQX448E message being issued.

CSQX548E: csect-name Messages sent to local dead-letter queue, channel channel-name reason=mqrc (mqrc-text)

- Follow the system programmer response.

CSQX788I: csect-name DNS lookup for address address using function 'func' took n seconds

- Follow the system programmer response.

CSQY225E: csect-name Queue manager is critically short of local storage above the bar - take action

- The queue manager is running critically short of virtual storage above the bar. Action should be taken to relieve the situation, and to avoid the possible abnormal termination of the queue manager.

CSQ5038I: csect-name Service task service-task has been unresponsive since hh.mm.ss.nnnnnn. Check for problems with Db2

- Follow the system programmer response.

Configurations à haute disponibilité

Si vous souhaitez utiliser vos gestionnaires de files d'attente IBM MQ dans une configuration à haute disponibilité, vous pouvez configurer vos gestionnaires de files d'attente pour qu'ils utilisent un gestionnaire à haute disponibilité, tel que PowerHA for AIX (anciennement HACMP) ou Microsoft Cluster Service (MSCS), ou avec des gestionnaires de files d'attente multi-instance IBM MQ. Sur les systèmes Linux, vous pouvez également déployer des gestionnaires de files d'attente de données répliquées (RDQM), qui utilisent un groupe basé sur un quorum pour assurer la haute disponibilité. Une autre option, Native HA, est destinée aux déploiements de conteneur.

MQ Appliance Une autre option pour une solution de haute disponibilité ou de reprise après incident consiste à déployer une paire de dispositifs IBM MQ. Voir [High Availability](#) et [Disaster Recovery](#) dans la documentation IBM MQ Appliance.

Vous devez connaître les définitions de configuration suivantes:

Clusters de gestionnaires de files d'attente

Groupes de deux ou plusieurs gestionnaires de files d'attente sur un ou plusieurs ordinateurs, fournissant une interconnexion automatique et permettant aux files d'attente d'être partagées entre eux pour l'équilibrage de charge et la redondance. A partir de la IBM WebSphere MQ 7.1, la reprise sur incident de cluster réexécute les opérations qui ont généré des problèmes jusqu'à ce que les problèmes soient résolus.

Clusters à haute disponibilité

Les clusters à haute disponibilité sont des groupes de deux ou plusieurs ordinateurs et ressources, tels que des disques et des réseaux, connectés ensemble et configurés de telle sorte que, en cas de défaillance, un gestionnaire de haute disponibilité, tel que HACMP (AIX and Linux) ou MSCS (Windows), effectue une *reprise en ligne*. La reprise en ligne transfère les données d'état des applications de l'ordinateur défaillant vers un autre ordinateur du cluster et y lance à nouveau leur opération. Cela offre une haute disponibilité des services exécutés dans le cluster à haute disponibilité. La relation entre les clusters IBM MQ et les clusters à haute disponibilité est décrite dans «[Relation entre les clusters à haute disponibilité et les clusters de gestionnaires de files d'attente](#)», à la page 518.

Gestionnaires de files d'attente multi-instance

Instances du même gestionnaire de files d'attente configurées sur deux ordinateurs ou plus. En démarrant plusieurs instances, une instance devient l'instance active et les autres instances deviennent des instances de secours. Si l'instance active échoue, une instance de secours s'exécutant sur un autre ordinateur prend automatiquement le relais. Vous pouvez utiliser des gestionnaires de files d'attente multi-instance pour configurer vos propres systèmes de messagerie à haute

disponibilité basés sur IBM MQ, sans avoir besoin d'une technologie de cluster telle que HACMP ou MSCS. Les clusters à haute disponibilité et les gestionnaires de files d'attente multi-instance sont d'autres moyens de rendre les gestionnaires de files d'attente hautement disponibles. Ne les combinez pas en plaçant un gestionnaire de files d'attente multi-instance dans un cluster à haute disponibilité.

Gestionnaires de files d'attente de données répliquées à haute disponibilité (RDQM à haute disponibilité)

Instances du même gestionnaire de files d'attente configuré sur chaque noeud d'un groupe de trois serveurs Linux . L'une des trois instances est l'instance active. Les données du gestionnaire de files d'attente actif sont répliquées de manière synchrone sur les deux autres instances, de sorte que l'une de ces instances peut prendre le relais en cas de défaillance. Le regroupement des serveurs est contrôlé par Pacemaker et la réplication par DRBD.

Gestionnaires de files d'attente de données répliquées pour la reprise après incident (DR RDQM)

Un gestionnaire de files d'attente s'exécute sur un noeud principal sur un site, avec une instance secondaire de ce gestionnaire de files d'attente située sur un noeud de reprise sur un autre site. Les données sont répliquées entre l'instance principale et l'instance secondaire, et si le noeud principal est perdu pour une raison quelconque, l'instance secondaire peut être transformée en instance principale et démarrée. Les deux noeuds doivent être des serveurs Linux . La réplication est contrôlée par DRBD.

Reprise après incident / gestionnaires de files d'attente de données répliquées à haute disponibilité (DR/HA RDQM)

Vous pouvez configurer un gestionnaire de files d'attente de données répliquées (RDQM) qui s'exécute sur un groupe de haute disponibilité sur un site, mais qui peut basculer sur un autre groupe de haute disponibilité sur un autre site en cas de sinistre qui rend le premier groupe indisponible. On appelle ce type de gestionnaire de files d'attente un gestionnaire de files d'attente de données répliquées de reprise après incident/à haute disponibilité (DR/HA).

CP4I

Native HA

Native HA est une solution à haute disponibilité destinée aux déploiements de conteneur d'IBM MQ. Native HA utilise la réplication de journal pour maintenir à jour trois instances d'un gestionnaire de files d'attente s'exécutant sur des noeuds différents. Une instance est active à la fois et traite les messages. Le gestionnaire de files d'attente actif envoie ses mises à jour de journal aux deux autres instances pour les maintenir à jour. Si l'instance active échoue, l'une des instances de réplique prend automatiquement le rôle actif.

Différences entre les gestionnaires de files d'attente multi-instance et les clusters à haute disponibilité

Les gestionnaires de files d'attente multi-instance et les clusters à haute disponibilité sont d'autres moyens d'assurer la haute disponibilité de vos gestionnaires de files d'attente. Voici quelques points qui mettent en évidence les différences entre les deux approches.

Les gestionnaires de files d'attente multi-instance incluent les fonctions suivantes:

- Prise en charge de la reprise en ligne de base intégrée à IBM MQ
- Reprise en ligne plus rapide que le cluster haute disponibilité
- Configuration et fonctionnement simples
- Intégration à IBM MQ Explorer

Les limitations des gestionnaires de files d'attente multi-instance sont les suivantes:

- Stockage en réseau haute performance et hautement disponible obligatoire
- Configuration réseau plus complexe car le gestionnaire de files d'attente modifie l'adresse IP en cas de basculement

Les clusters à haute disponibilité incluent les fonctions suivantes:

- Possibilité de coordonner plusieurs ressources, telles qu'un serveur d'applications ou une base de données
- Options de configuration plus flexibles incluant des clusters comportant plus de deux noeuds
- Peut basculer plusieurs fois sans intervention de l'opérateur
- Reprise de l'adresse IP du gestionnaire de files d'attente dans le cadre de la reprise en ligne


Les limitations des clusters à haute disponibilité sont les suivantes:

- Des compétences et des achats de produits supplémentaires sont requis
- Les disques pouvant être commutés entre les noeuds du cluster sont requis
- La configuration des clusters à haute disponibilité est relativement complexe
- La reprise en ligne est assez lente d'un point de vue historique, mais les récents produits de cluster à haute disponibilité l'améliorent
- Des basculements inutiles peuvent se produire si les scripts utilisés pour surveiller les ressources, telles que les gestionnaires de files d'attente, présentent des lacunes.

Relation entre les clusters à haute disponibilité et les clusters de gestionnaires de files d'attente

Les clusters de gestionnaires de files d'attente fournissent un équilibrage de charge des messages entre les instances disponibles des files d'attente de clusters de gestionnaires de files d'attente. Cela offre une disponibilité supérieure à celle d'un gestionnaire de files d'attente unique car, à la suite d'une défaillance d'un gestionnaire de files d'attente, les applications de messagerie peuvent toujours envoyer des messages à des instances d'une file d'attente de cluster de gestionnaire de files d'attente et y accéder. Toutefois, bien que les clusters de gestionnaires de files d'attente routent automatiquement les nouveaux messages vers les gestionnaires de files d'attente disponibles dans un cluster, les messages actuellement mis en file d'attente sur un gestionnaire de files d'attente non disponible ne sont pas disponibles tant que ce gestionnaire de files d'attente n'est pas redémarré. Pour cette raison, les clusters de gestionnaires de files d'attente ne fournissent pas à eux seuls la haute disponibilité de toutes les données de message, ni la détection automatique de la défaillance du gestionnaire de files d'attente et le déclenchement automatique du redémarrage ou de la reprise en ligne du gestionnaire de files d'attente. Les clusters à haute disponibilité (HA) fournissent ces fonctions. Les deux types de cluster peuvent être utilisés ensemble avec un bon effet. Pour une présentation des clusters de gestionnaires de files d'attente, voir [Conception de clusters](#).

Concepts associés

 Haute disponibilité pour le IBM MQ Advanced container

Clusters à haute disponibilité sur AIX and Linux

Vous pouvez utiliser IBM MQ avec un cluster à haute disponibilité sur les plateformes AIX and Linux : par exemple, PowerHA for AIX (anciennement HACMP), Veritas Cluster Server, HP Serviceguard ou un cluster Red Hat Enterprise Linux avec Red Hat Cluster Suite.

Cette section présente «Configurations de cluster à haute disponibilité», à la page 518, la relation des clusters à haute disponibilité avec les clusters de gestionnaires de files d'attente, «IBM MQ clients», à la page 519 et «IBM MQ fonctionnant dans un cluster à haute disponibilité», à la page 520, et vous guide tout au long des étapes et fournit des exemples de script que vous pouvez adapter pour configurer les gestionnaires de files d'attente avec un cluster à haute disponibilité.

Reportez-vous à la documentation du cluster à haute disponibilité spécifique à votre environnement pour obtenir de l'aide sur les étapes de configuration décrites dans cette section.

Configurations de cluster à haute disponibilité

Dans cette section, le terme *noeud* est utilisé pour désigner l'entité qui exécute un système d'exploitation et le logiciel à haute disponibilité ; "ordinateur", "système" ou "machine" ou "partition" ou "blade" peut

être considéré comme synonyme de cette utilisation. Vous pouvez utiliser IBM MQ pour vous aider à définir des configurations de secours ou de reprise, y compris la reprise mutuelle où tous les noeuds de cluster exécutent la charge de travail IBM MQ .

Une configuration de *secours* est la configuration de cluster à haute disponibilité la plus basique dans laquelle un noeud effectue un travail alors que l'autre noeud agit uniquement en tant que noeud de secours. Le noeud de secours n'effectue pas de travail et est appelé inactif ; cette configuration est parfois appelée *cold standby*. Une telle configuration nécessite un haut degré de redondance matérielle. Pour économiser du matériel, il est possible d'étendre cette configuration pour avoir plusieurs noeuds worker avec un seul noeud de secours. Cela signifie que le noeud de secours peut prendre en charge le travail de n'importe quel autre noeud worker. Cette configuration est toujours appelée configuration de secours et parfois configuration "N+1".

Une configuration de *reprise* est une configuration plus avancée dans laquelle tous les noeuds effectuent des travaux et des travaux critiques peuvent être repris en cas de défaillance d'un noeud.

Une configuration de *reprise unilatérale* est une configuration dans laquelle un noeud de secours effectue un travail supplémentaire, non critique et non déplaçable. Cette configuration est similaire à une configuration de secours, mais avec un travail (non critique) effectué par le noeud de secours.

Une configuration de *reprise mutuelle* est une configuration dans laquelle tous les noeuds effectuent un travail à haute disponibilité (déplaçable). Ce type de configuration de cluster à haute disponibilité est également parfois appelé "Active / Active" pour indiquer que tous les noeuds traitent activement la charge de travail critique.

Avec la configuration de secours étendue ou l'une ou l'autre des configurations de reprise, il est important de prendre en compte la charge maximale qui peut être placée sur un noeud qui peut prendre en charge le travail d'autres noeuds. Un tel noeud doit posséder une capacité suffisante pour maintenir un niveau de performance acceptable.

Relation entre les clusters à haute disponibilité et les clusters de gestionnaires de files d'attente

Les clusters de gestionnaires de files d'attente réduisent l'administration et assurent l'équilibrage de charge des messages entre les instances des files d'attente de clusters de gestionnaires de files d'attente. Ils offrent également une disponibilité supérieure à celle d'un seul gestionnaire de files d'attente car, à la suite d'une défaillance d'un gestionnaire de files d'attente, les applications de messagerie peuvent toujours accéder aux instances survivantes d'une file d'attente de cluster de gestionnaire de files d'attente. Toutefois, les clusters de gestionnaires de files d'attente ne fournissent pas à eux seuls la détection automatique de la défaillance du gestionnaire de files d'attente et le déclenchement automatique du redémarrage ou de la reprise en ligne du gestionnaire de files d'attente. Les clusters à haute disponibilité fournissent ces fonctions. Les deux types de cluster peuvent être utilisés ensemble avec un bon effet.

IBM MQ clients

Les clients IBM MQ qui communiquent avec un gestionnaire de files d'attente pouvant faire l'objet d'un redémarrage ou d'une reprise doivent être écrits pour tolérer une connexion interrompue et doivent tenter à plusieurs reprises de se reconnecter. IBM MQ inclut des fonctions dans le traitement de la table de définition de canal du client (CCDT) qui aident à la disponibilité des connexions et à l'équilibrage de la charge de travail ; toutefois, ces fonctions ne sont pas directement pertinentes lors de l'utilisation d'un système de reprise en ligne.

La fonctionnalité transactionnelle permet à un IBM MQ MQI client de participer à des transactions en deux phases, à condition que le client soit connecté au même gestionnaire de files d'attente. La fonctionnalité transactionnelle ne peut pas utiliser de techniques, telles qu'un équilibreur de charge IP, pour effectuer une sélection dans une liste de gestionnaires de files d'attente. Lorsque vous utilisez un produit à haute disponibilité, un gestionnaire de files d'attente conserve son identité (nom et adresse) quel que soit le noeud sur lequel il s'exécute, afin que la fonctionnalité transactionnelle puisse être utilisée avec les gestionnaires de files d'attente sous contrôle de haute disponibilité.

IBM MQ fonctionnant dans un cluster à haute disponibilité

Tous les clusters à haute disponibilité ont le concept d'une unité de reprise en ligne. Il s'agit d'un ensemble de définitions qui contient toutes les ressources qui constituent le service à haute disponibilité. L'unité de reprise en ligne inclut le service lui-même et toutes les autres ressources dont il dépend.

Les solutions à haute disponibilité utilisent des termes différents pour une unité de reprise en ligne:

- Sous PowerHA for AIX , l'unité de reprise en ligne est appelée *groupe de ressources*.
- Sur Veritas Cluster Server, il s'agit d'un *groupe de services*.
- Sur Serviceguard, il est appelé *package*.

Cette rubrique utilise le terme *groupe de ressources* pour désigner une unité de reprise en ligne.

La plus petite unité de reprise en ligne pour IBM MQ est un gestionnaire de files d'attente. Généralement, le groupe de ressources contenant le gestionnaire de files d'attente contient également des disques partagés dans un groupe de volumes ou un groupe de disques réservé exclusivement à l'utilisation par le groupe de ressources, ainsi que l'adresse IP utilisée pour la connexion au gestionnaire de files d'attente. Il est également possible d'inclure d'autres ressources IBM MQ , telles qu'un programme d'écoute ou un moniteur de déclenchement dans le même groupe de ressources, en tant que ressources distinctes ou sous le contrôle du gestionnaire de files d'attente lui-même.

Un gestionnaire de files d'attente à utiliser dans un cluster à haute disponibilité doit avoir ses données et ses journaux sur des disques partagés entre les noeuds du cluster. Le cluster à haute disponibilité garantit qu'un seul noeud du cluster à la fois peut écrire sur les disques. Le cluster à haute disponibilité peut utiliser un script de surveillance pour surveiller l'état du gestionnaire de files d'attente.

Il est possible d'utiliser un disque partagé unique pour les données et les journaux associés au gestionnaire de files d'attente. Toutefois, il est normal d'utiliser des systèmes de fichiers partagés distincts afin qu'ils puissent être dimensionnés et optimisés indépendamment.

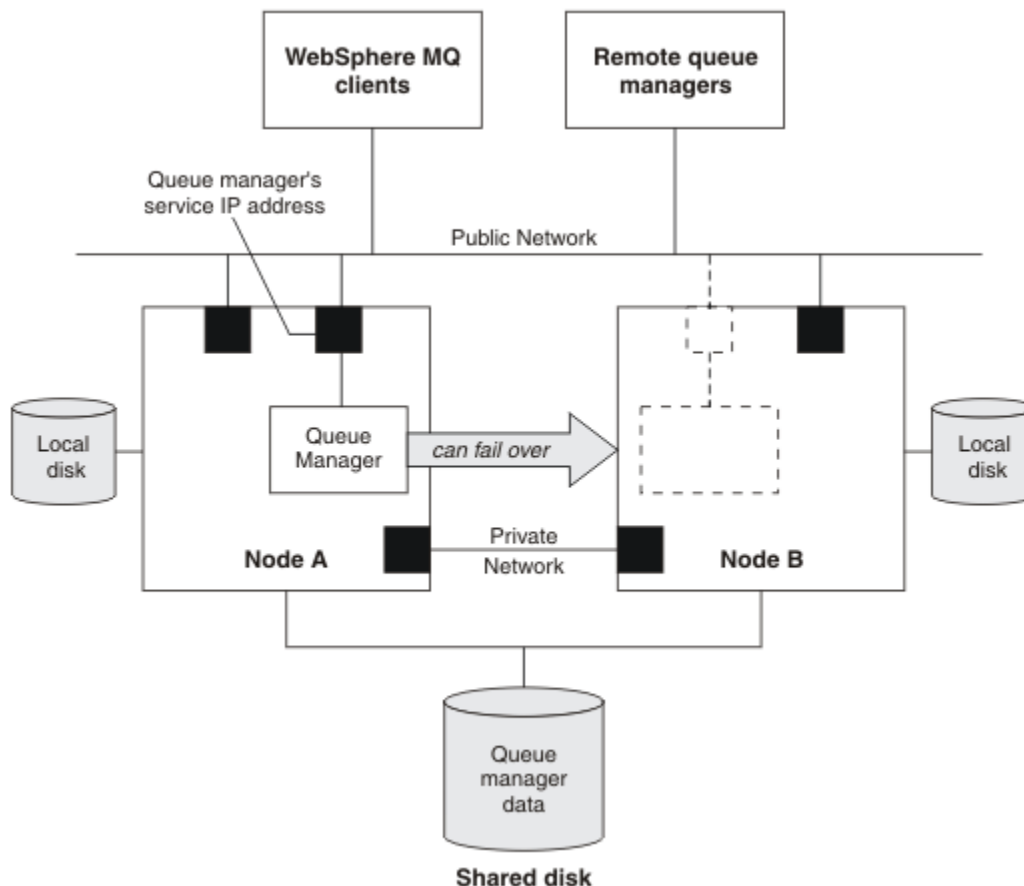


Figure 69. HA cluster

La [Figure 1](#) illustre un cluster à haute disponibilité avec deux noeuds. Le cluster à haute disponibilité gère la disponibilité d'un gestionnaire de files d'attente qui a été défini dans un groupe de ressources. Il s'agit d'une configuration active / passive ou de secours (cold standby), car un seul noeud, le noeud A, exécute actuellement un gestionnaire de files d'attente. Le gestionnaire de files d'attente a été créé avec ses données et ses fichiers journaux sur un disque partagé. Le gestionnaire de files d'attente possède une adresse IP de service qui est également gérée par le cluster à haute disponibilité. Le gestionnaire de files d'attente dépend du disque partagé et de son adresse IP de service. Lorsque le cluster à haute disponibilité fait échouer le gestionnaire de files d'attente du noeud A vers le noeud B, il déplace d'abord les ressources dépendantes du gestionnaire de files d'attente sur le noeud B, puis démarre le gestionnaire de files d'attente.

Si le cluster à haute disponibilité contient plusieurs gestionnaires de files d'attente, la configuration de votre cluster à haute disponibilité peut entraîner l'exécution de plusieurs gestionnaires de files d'attente sur le même noeud après une reprise en ligne. Chaque gestionnaire de files d'attente du cluster à haute disponibilité doit se voir affecter son propre numéro de port, qu'il utilise sur le noeud de cluster qu'il se trouve être actif à un moment donné.

En règle générale, le cluster à haute disponibilité s'exécute en tant que superutilisateur. IBM MQ s'exécute en tant qu'utilisateur mqm. L'administration de IBM MQ est accordée aux membres du groupe mqm. Vérifiez que l'utilisateur et le groupe mqm existent sur tous les noeuds de cluster à haute disponibilité. L'ID utilisateur et l'ID groupe doivent être cohérents dans le cluster. L'administration de IBM MQ par l'utilisateur root n'est pas autorisée ; les scripts qui démarrent, arrêtent ou surveillent les scripts doivent passer à l'utilisateur mqm.

Remarque : IBM MQ doit être installé correctement sur tous les noeuds ; vous ne pouvez pas partager les fichiers exécutables du produit.

Linux

AIX

Configuration de disques partagés sur AIX and Linux

Un gestionnaire de files d'attente IBM MQ dans un cluster à haute disponibilité requiert que les fichiers de données et les fichiers journaux se trouvent dans des systèmes de fichiers distants nommés communs sur un disque partagé.

Pourquoi et quand exécuter cette tâche

La [Figure 1](#) présente un agencement possible pour un gestionnaire de files d'attente dans un cluster à haute disponibilité. Les répertoires de données et de journaux du gestionnaire de files d'attente se trouvent sur le disque partagé qui est monté sur /MQHA/QM1. Ce disque est commuté entre les noeuds du cluster à haute disponibilité lors de la reprise en ligne afin que les données soient disponibles partout où le gestionnaire de files d'attente est redémarré. Le fichier `mqm.ini` contient une strophe pour le gestionnaire de files d'attente QM1. La section Journal du fichier `qm.ini` contient une valeur pour `LogPath`.

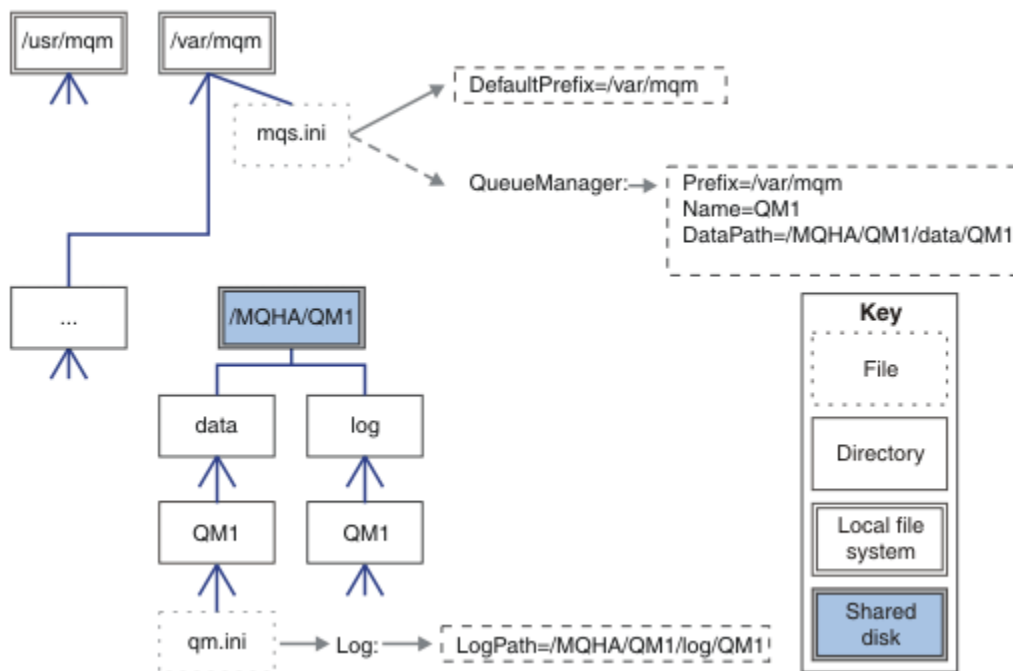


Figure 70. Répertoires data et log partagés

Procédure

1. Choisissez les noms des points de montage pour les systèmes de fichiers du gestionnaire de files d'attente.
Par exemple, `/MQHA/qmgrname/data` pour les fichiers de données du gestionnaire de files d'attente et `/MQHA/qmgrname/log` pour ses fichiers journaux.
2. Créez un groupe de volumes (ou de disques) destiné à contenir les données et les fichiers journaux du gestionnaire de files d'attente.
Ce groupe de volumes est géré par le cluster à haute disponibilité (HA) dans le même groupe de ressources que le gestionnaire de files d'attente.
3. Créez les systèmes de fichiers pour les données et les fichiers journaux du gestionnaire de files d'attente dans le groupe de volumes.
4. Pour chaque noeud, créez les points de montage pour les systèmes de fichiers et assurez-vous que les systèmes de fichiers peuvent être montés.
L'utilisateur `mqm` doit être propriétaire des points de montage.

Linux AIX **Création d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux**

La première étape de l'utilisation d'un gestionnaire de files d'attente dans un cluster à haute disponibilité consiste à créer le gestionnaire de files d'attente sur l'un des noeuds.

Pourquoi et quand exécuter cette tâche

Pour créer un gestionnaire de files d'attente à utiliser dans un cluster à haute disponibilité, vous devez d'abord sélectionner l'un des noeuds du cluster sur lequel créer le gestionnaire de files d'attente, puis effectuer les étapes suivantes sur ce noeud.

Procédure

1. Montez les systèmes de fichiers du gestionnaire de files d'attente sur le noeud.
2. Créez le gestionnaire de files d'attente à l'aide de la commande `crtmqm`.

Exemple :

```
crtmqm -md /MQHA/qmgrname/data -ld /MQHA/qmgrname/log qmgrname
```

3. Démarrez le gestionnaire de files d'attente manuellement à l'aide de la commande **strmqm** .
4. Effectuez la configuration initiale du gestionnaire de files d'attente, par exemple en créant des files d'attente et des canaux, et en définissant le gestionnaire de files d'attente de sorte qu'il démarre automatiquement un programme d'écoute au démarrage du gestionnaire de files d'attente.
5. Arrêtez le gestionnaire de files d'attente à l'aide de la commande **endmqm** .
6. Utilisez la commande **dspmqlnf** pour afficher la commande **addmqinf** :

```
dspmqlnf -o command qmgrname
```

où qmgrname est le nom du gestionnaire de files d'attente.

Pour plus d'informations sur l'utilisation de la commande **addmqinf** , voir «Ajout d'une configuration de gestionnaire de files d'attente à d'autres noeuds de cluster à haute disponibilité sur AIX and Linux», à la page 523.

La commande **addmqinf** s'affiche de manière similaire à l'exemple suivant:

```
addmqinf -sQueueManager -vName=qmgrname -vDirectory=qmgrname \  
-vPrefix=/var/mqm -vDataPath=/MQHA/qmgrname/data/qmgrname
```

7. Prenez soin de noter la commande affichée.
8. Démontez les systèmes de fichiers du gestionnaire de files d'attente.

Que faire ensuite

Vous êtes maintenant prêt à effectuer les étapes décrites dans «Ajout d'une configuration de gestionnaire de files d'attente à d'autres noeuds de cluster à haute disponibilité sur AIX and Linux», à la page 523.

Ajout d'une configuration de gestionnaire de files d'attente à d'autres noeuds de cluster à haute disponibilité sur AIX and Linux

Vous devez ajouter les informations de configuration du gestionnaire de files d'attente aux autres noeuds du cluster à haute disponibilité.

Avant de commencer

Avant d'effectuer cette tâche, vous devez avoir effectué les étapes de la rubrique «Création d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux», à la page 522.

Après avoir créé le gestionnaire de files d'attente, vous devez ajouter les informations de configuration du gestionnaire de files d'attente à chacun des autres noeuds du cluster à haute disponibilité en procédant comme suit sur chacun des autres noeuds.

Pourquoi et quand exécuter cette tâche

Lorsque vous créez un gestionnaire de files d'attente à utiliser dans un cluster à haute disponibilité, vous devez d'abord sélectionner l'un des noeuds du cluster sur lequel vous souhaitez créer le gestionnaire de files d'attente, comme décrit dans «Création d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux», à la page 522.

Procédure

1. Montez les systèmes de fichiers du gestionnaire de files d'attente.
2. Ajoutez les informations de configuration du gestionnaire de files d'attente au noeud.
Il existe deux façons d'ajouter les informations de configuration:
 - En éditant `/var/mqm/mqs.ini` directement.

- En exécutant la commande **addmqinf** qui a été affichée par la commande **dspmqlnf** à l'étape 6 de la rubrique «Création d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux», à la page 522.
3. Démarrez et arrêtez le gestionnaire de files d'attente pour vérifier la configuration.

Les commandes utilisées pour démarrer et arrêter le gestionnaire de files d'attente doivent être émises à partir de la même installation IBM MQ que la commande **addmqinf**. Pour démarrer et arrêter le gestionnaire de files d'attente à partir d'une installation différente de celle qui est actuellement associée au gestionnaire de files d'attente, vous devez d'abord définir l'installation associée au gestionnaire de files d'attente à l'aide de la commande **setmqm**. Pour plus d'informations, voir [setmqm](#).
 4. Démontez les systèmes de fichiers du gestionnaire de files d'attente.

Linux AIX **Exemples de scripts shell pour le démarrage d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux**

Le gestionnaire de files d'attente est représenté dans le cluster à haute disponibilité en tant que ressource. Le cluster à haute disponibilité doit pouvoir démarrer et arrêter le gestionnaire de files d'attente. Dans la plupart des cas, vous pouvez utiliser un script shell pour démarrer le gestionnaire de files d'attente. Vous devez rendre ces scripts disponibles au même emplacement sur tous les noeuds du cluster, en utilisant un système de fichiers réseau ou en les copiant sur chacun des disques locaux.

Remarque : Avant de redémarrer un gestionnaire de files d'attente défaillant, vous devez déconnecter vos applications de cette instance du gestionnaire de files d'attente. Si vous ne le faites pas, le gestionnaire de files d'attente risque de ne pas redémarrer correctement.

Des exemples de scripts shell appropriés sont donnés ici. Vous pouvez les adapter à vos besoins et les utiliser pour démarrer le gestionnaire de files d'attente sous le contrôle de votre cluster à haute disponibilité.

Le script shell suivant montre comment passer de l'utilisateur de cluster à haute disponibilité à l'utilisateur mqm afin que le gestionnaire de files d'attente puisse être démarré:

```
#!/bin/ksh
# A simple wrapper script to switch to the mqm user.
su mqm -c name_of_your_script $*
```

Le script shell suivant est un exemple de démarrage d'un gestionnaire de files d'attente sans faire d'hypothèses sur l'état en cours du gestionnaire de files d'attente. Notez qu'il utilise une méthode extrêmement brutale d'arrêt des processus appartenant au gestionnaire de files d'attente:

```
#!/bin/ksh
#
# This script robustly starts the queue manager.
#
# The script must be run by the mqm user.
#
# The only argument is the queue manager name. Save it as QM variable
QM=$1

if [ -z "$QM" ]
then
  echo "ERROR! No queue manager name supplied"
  exit 1
fi

# End any queue manager processes which might be running.

srchstr="(|-m)$QM *.*$"
for process in amqzmuc0 amqzma0 amqfcbx0 amqfcpub amqpcsea amqzlaa0 \
               amqzlsa0 runmqchi runmqslr amqcrista amqirmfa amqimppa \
               amqzfuma amqzmuf0 amqzmur0 amqzmgr0
do
  ps -ef | tr "\t" " " | grep $process | grep -v grep | \
  egrep "$srchstr" | awk '{print $2}' | \
```

```

done      xargs kill -9 > /dev/null 2>&1

# It is now safe to start the queue manager.
# The stimqm command does not use the -x flag.
stimqm ${QM}

```

Vous pouvez modifier le script pour démarrer d'autres programmes associés.

Linux AIX **Exemple de script shell pour l'arrêt d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux**

Dans la plupart des cas, vous pouvez utiliser un script shell pour arrêter un gestionnaire de files d'attente. Des exemples de scripts shell appropriés sont donnés ici. Vous pouvez les adapter à vos besoins et les utiliser pour arrêter le gestionnaire de files d'attente sous le contrôle de votre cluster à haute disponibilité.

Le script suivant est un exemple d'arrêt immédiat d'un gestionnaire de files d'attente sans faire d'hypothèses sur l'état en cours du gestionnaire de files d'attente. Le script doit être exécuté par l'utilisateur mqm. Il peut donc s'avérer nécessaire d'encapsuler ce script dans un script shell pour passer de l'utilisateur du cluster à haute disponibilité à mqm. (Un exemple de script shell est fourni dans «Exemples de scripts shell pour le démarrage d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux», à la page 524.)

```

#!/bin/ksh
#
# The script ends the QM by using two phases, initially trying an immediate
# end with a time-out and escalating to a forced stop of remaining
# processes.
#
# The script must be run by the mqm user.
#
# There are two arguments: the queue manager name and a timeout value.
QM=$1
TIMEOUT=$2

if [ -z "$QM" ]
then
    echo "ERROR! No queue manager name supplied"
    exit 1
fi

if [ -z "$TIMEOUT" ]
then
    echo "ERROR! No timeout specified"
    exit 1
fi

for severity in immediate brutal
do
    # End the queue manager in the background to avoid
    # it blocking indefinitely. Run the TIMEOUT timer
    # at the same time to interrupt the attempt, and try a
    # more forceful version. If the brutal version fails,
    # nothing more can be done here.

    echo "Attempting ${severity} end of queue manager '${QM}'"
    case $severity in

immediate)
        # Minimum severity of endmqm is immediate which severs connections.
        # HA cluster should not be delayed by clients
        endmqm -i ${QM} &
        ;;

brutal)
        # This is a forced means of stopping queue manager processes.

        srchstr="(|-m)$QM *.*$"
        for process in amqzmuc0 amqzxma0 amqfcxba amqfcpub amqpcsea amqzlaa0 \
            amqzlsa0 runmqchi runmqlsr amqcrsta amqrrmf0 amqrmppa \
            amqzfuma amqmuf0 amqzmur0 amqzmgr0
        do
            ps -ef | tr "\t" " " | grep $process | grep -v grep | \
                egrep "$srchstr" | awk '{print $2}' | \

```

```

        xargs kill -9 > /dev/null 2>&1
    done

esac

TIMED_OUT=yes
SECONDS=0
while (( $SECONDS < ${TIMEOUT} ))
do
    TIMED_OUT=yes
    i=0
    while [ $i -lt 5 ]
    do
        # Check for execution controller termination
        srchstr="( |-m)$QM *.*$"
        cnt=`ps -ef | tr "\t" " " | grep amqzma0 | grep -v grep | \
            egrep "$srchstr" | awk '{print $2}' | wc -l`
        i=`expr $i + 1`
        sleep 1
        if [ $cnt -eq 0 ]
        then
            TIMED_OUT=no
            break
        fi
    done

    if [ ${TIMED_OUT} = "no" ]
    then
        break
    fi

    echo "Waiting for ${severity} end of queue manager '${QM}'"
    sleep 1
done # timeout loop

if [ ${TIMED_OUT} = "yes" ]
then
    continue      # to next level of urgency
else
    break         # queue manager is ended, job is done
fi

done # next phase

```

Remarque : Selon les processus en cours d'exécution pour un gestionnaire de files d'attente spécifique, la liste des processus de gestionnaire de files d'attente inclus dans ce script peut ne pas être une liste complète ou peut inclure plus de processus que les processus en cours d'exécution pour ce gestionnaire de files d'attente:

```

for process in amqzmuc0 amqzma0 amqfcxba amqfipub amqpcsea amqzlaa0 \
    amqzlsa0 runmqchi runmqlsr amqcrista amqirmfa amqimppa \
    amqzfuma amqmuf0 amqzmur0 amqzmgr0

```

Un processus peut être inclus dans ou exclu de cette liste en fonction de la fonction configurée et des processus en cours d'exécution pour un gestionnaire de files d'attente spécifique. Pour obtenir la liste complète des processus et des informations sur l'arrêt des processus dans un ordre spécifique, voir [Arrêt manuel d'un gestionnaire de files d'attente sous UNIX et Linux](#).

Linux

AIX

Surveillance d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux

Il est habituel de permettre au cluster à haute disponibilité de surveiller périodiquement l'état du gestionnaire de files d'attente. Dans la plupart des cas, vous pouvez utiliser un script shell pour cela. Des exemples de scripts shell appropriés sont donnés ici. Vous pouvez personnaliser ces scripts en fonction de vos besoins et les utiliser pour effectuer des vérifications de surveillance supplémentaires spécifiques à votre environnement.

Il est possible que plusieurs installations de IBM MQ coexistent sur un système. Pour plus d'informations sur les installations multiples, voir [Installations multiples](#). Si vous prévoyez d'utiliser le script de surveillance sur plusieurs installations, vous devrez peut-être effectuer des étapes supplémentaires. Si vous disposez d'une installation principale, vous n'avez pas besoin de spécifier le

`MQ_INSTALLATION_PATH` pour utiliser le script. Sinon, procédez comme suit pour vous assurer que le `MQ_INSTALLATION_PATH` est correctement identifié:

1. Utilisez la commande **`crtmqenv`** à partir d'une installation IBM MQ pour identifier le `MQ_INSTALLATION_PATH` approprié pour un gestionnaire de files d'attente:

```
crtmqenv -m qmname
```

Cette commande renvoie la valeur `MQ_INSTALLATION_PATH` correcte pour le gestionnaire de files d'attente spécifié par `qmname`.

2. Exécutez le script de surveillance avec les paramètres `qmname` et `MQ_INSTALLATION_PATH` appropriés.

Remarque : PowerHA for AIX ne permet pas de fournir un paramètre au programme de surveillance du gestionnaire de files d'attente. Vous devez créer un programme de surveillance distinct pour chaque gestionnaire de files d'attente, qui encapsule le nom du gestionnaire de files d'attente. Voici un exemple de script utilisé sur AIX pour encapsuler le nom du gestionnaire de files d'attente:

```
#!/bin/ksh
su mqm -c name_of_monitoring_script qmname MQ_INSTALLATION_PATH
```

où `MQ_INSTALLATION_PATH` est un paramètre facultatif qui spécifie le chemin d'accès à l'installation de IBM MQ à laquelle le gestionnaire de files d'attente `qmname` est associé.

Le script suivant n'est pas robuste à la possibilité que **`runmqsc`** se bloque. En règle générale, les clusters à haute disponibilité traitent un script de surveillance de blocage comme un échec et sont eux-mêmes robustes face à cette possibilité.

Le script tolère cependant que le gestionnaire de files d'attente soit à l'état de démarrage. En effet, il est courant que le cluster à haute disponibilité démarre la surveillance du gestionnaire de files d'attente dès qu'il l'a démarré. Certains clusters à haute disponibilité font la distinction entre une phase de démarrage et une phase d'exécution pour les ressources, mais il est nécessaire de configurer la durée de la phase de démarrage. Etant donné que la durée de démarrage d'un gestionnaire de files d'attente dépend de la quantité de travail qu'il doit effectuer, il est difficile de choisir une durée maximale de démarrage d'un gestionnaire de files d'attente. Si vous choisissez une valeur trop basse, le cluster à haute disponibilité suppose à tort que le gestionnaire de files d'attente a échoué lorsqu'il n'a pas terminé son démarrage. Cela pourrait entraîner une suite sans fin de basculements.

Ce script doit être exécuté par l'utilisateur `mqm` ; il peut donc être nécessaire d'encapsuler ce script dans un script shell pour passer de l'utilisateur du cluster à haute disponibilité à `mqm` (un exemple de script shell est fourni dans [«Exemples de scripts shell pour le démarrage d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux»](#), à la page 524):

```
#!/bin/ksh
#
# This script tests the operation of the queue manager.
#
# An exit code is generated by the runmqsc command:
# 0 => Either the queue manager is starting or the queue manager is running and responds.
#     Either is OK.
# >0 => The queue manager is not responding and not starting.
#
# This script must be run by the mqm user.
QM=$1
MQ_INSTALLATION_PATH=$2

if [ -z "$QM" ]
then
    echo "ERROR! No queue manager name supplied"
    exit 1
fi

if [ -z "$MQ_INSTALLATION_PATH" ]
then
    # No path specified, assume system primary install or MQ level < 7.1.0.0
    echo "INFO: Using shell default value for MQ_INSTALLATION_PATH"
```

```

else
  echo "INFO: Prefixing shell PATH variable with $MQ_INSTALLATION_PATH/bin"
  PATH=$MQ_INSTALLATION_PATH/bin:$PATH
fi

# Test the operation of the queue manager. Result is 0 on success, non-zero on error.
echo "ping qmgr" | runmqsc ${QM} > /dev/null 2>&1
pingresult=$?

if [ $pingresult -eq 0 ]
then # ping succeeded

  echo "Queue manager '${QM}' is responsive"
  result=0

else # ping failed

  # Don't condemn the queue manager immediately, it might be starting.
  srchstr="(|-m)$QM *.*$"
  cnt=`ps -ef | tr "\t" " " | grep stmqm | grep "$srchstr" | grep -v grep \
    | awk '{print $2}' | wc -l`
  if [ $cnt -gt 0 ]
  then
    # It appears that the queue manager is still starting up, tolerate
    echo "Queue manager '${QM}' is starting"
    result=0
  else
    # There is no sign of the queue manager starting
    echo "Queue manager '${QM}' is not responsive"
    result=$pingresult
  fi
fi

fi

exit $result

```

Linux

AIX

Placement du gestionnaire de files d'attente sous le contrôle de cluster à haute disponibilité sous AIX and Linux

Vous devez configurer le gestionnaire de files d'attente, sous le contrôle du cluster à haute disponibilité, avec l'adresse IP et les disques partagés du gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Pour placer le gestionnaire de files d'attente sous le contrôle du cluster à haute disponibilité, vous devez définir un groupe de ressources qui contiendra le gestionnaire de files d'attente et toutes ses ressources associées.

Procédure

1. Créez le groupe de ressources contenant le gestionnaire de files d'attente, le volume ou le groupe de disques du gestionnaire de files d'attente et l'adresse IP du gestionnaire de files d'attente.
L'adresse IP est une adresse IP virtuelle et non l'adresse IP de l'ordinateur.
2. Vérifiez que le cluster à haute disponibilité bascule correctement les ressources entre les noeuds du cluster et qu'il est prêt à contrôler le gestionnaire de files d'attente.

Linux

AIX

Suppression d'un gestionnaire de files d'attente de cluster à haute disponibilité sous AIX and Linux

Vous pouvez supprimer un gestionnaire de files d'attente d'un noeud qui n'est plus nécessaire pour exécuter le gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Pour supprimer le gestionnaire de files d'attente d'un noeud d'un cluster à haute disponibilité, vous devez supprimer ses informations de configuration.

Procédure

1. Retirez le noeud du cluster à haute disponibilité afin que le cluster à haute disponibilité ne tente plus d'activer le gestionnaire de files d'attente sur ce noeud.
2. Utilisez la commande **rmvmqinf** suivante pour supprimer les informations de configuration du gestionnaire de files d'attente:

```
rmvmqinf qmgrname
```

3. Facultatif : Pour supprimer complètement le gestionnaire de files d'attente, utilisez la commande **dltmqm**.

Important : N'oubliez pas que la suppression du gestionnaire de files d'attente à l'aide de la commande **dltmqm** supprime complètement les données et les fichiers journaux du gestionnaire de files d'attente.

Une fois que vous avez supprimé le gestionnaire de files d'attente, vous pouvez utiliser la commande **rmvmqinf** pour supprimer les informations de configuration restantes des autres noeuds.

Windows Prise en charge de Microsoft Cluster Service (MSCS)

Présentation et configuration de MSCS pour la prise en charge de la reprise en ligne des serveurs virtuels. MSCS est également appelé Windows Server Failover Clustering (WSFC).

Ces informations s'appliquent à IBM MQ for Windows uniquement.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

MSCS/WSFC vous permet de connecter des serveurs dans un cluster, ce qui offre une plus grande disponibilité des données et des applications et facilite la gestion du système. MSCS/WSFC peut détecter et récupérer automatiquement en cas de défaillance du serveur ou de l'application.

MSCS/WSFC prend en charge la reprise en ligne des serveurs virtuels, qui correspondent à des applications, des sites Web, des files d'attente d'impression ou des partages de fichiers (y compris, par exemple, leurs axes de disque, leurs fichiers et leurs adresses IP).

La *reprise en ligne* est le processus par lequel MSCS/WSFC détecte une défaillance dans une application sur un ordinateur du cluster et arrête l'application perturbée de manière ordonnée, transfère ses données d'état à l'autre ordinateur, puis lance à nouveau l'application sur ce dernier.

Pour plus d'informations sur la configuration et l'utilisation des clusters de reprise en ligne, voir les sous-rubriques.

Windows Présentation des clusters MSCS

Les clusters Microsoft Cluster Service (MSCS) sont des groupes de deux ou plusieurs ordinateurs, connectés entre eux et configurés de telle sorte 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 y relayant leur fonctionnement.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

«Configurations à haute disponibilité», à la page 516 contient une comparaison entre les clusters MSCS, les gestionnaires de files d'attente multi-instance et les clusters IBM MQ.

Dans cette section et ses rubriques subordonnées, le terme *cluster*, lorsqu'il est utilisé seul, **toujours** signifie un cluster MSCS. Il est distinct d'un cluster IBM MQ décrit ailleurs dans ce guide.

Un cluster à deux machines comprend deux ordinateurs (par exemple, A et B) qui sont connectés conjointement à un réseau pour l'accès client à l'aide d'une *adresse IP virtuelle*. Ces deux ordinateurs peuvent également être connectés l'un à l'autre par un ou plusieurs réseaux privés. A et B partagent au moins un disque pour les applications serveur qu'ils utilisent tous les deux. Il existe également un autre disque partagé, qui doit être une grappe redondante de disques indépendants (*RAID*) Niveau 1, pour

l'utilisation exclusive de MSCS ; il s'agit du disque *quorum* . MSCS surveille les deux ordinateurs afin de s'assurer du bon fonctionnement du matériel et des logiciels.

Dans une configuration simple comme celle-ci, les deux ordinateurs ont toutes les applications installées sur eux, mais seul l'ordinateur A s'exécute avec des applications en temps réel ; l'ordinateur B est juste en cours d'exécution et en attente. Si l'ordinateur A rencontre l'un ou l'autre des problèmes, MSCS arrête l'application perturbée de manière ordonnée, transfère ses données d'état à l'autre ordinateur et y lance à nouveau l'application. Il s'agit d'une *reprise en ligne*. Les applications peuvent être rendues *sensibles aux clusters* afin qu'elles puissent interagir pleinement avec MSCS et effectuer une reprise en ligne en toute sécurité.

Pour un exemple de configuration ordinaire d'un cluster de deux ordinateurs, voir [Figure 71](#), à la page 530.

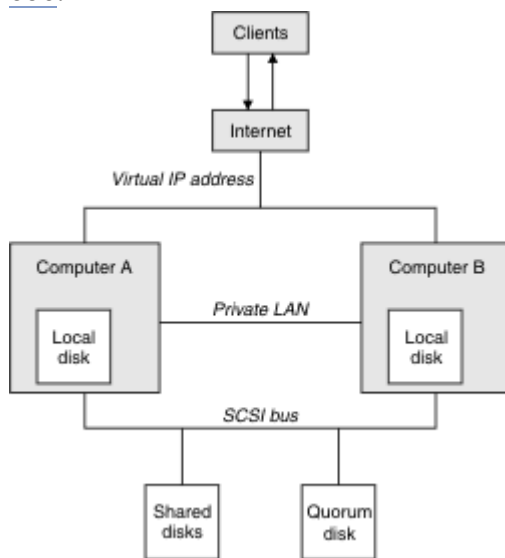


Figure 71. Cluster MSCS à deux ordinateurs

Chaque ordinateur peut accéder au disque partagé, mais un seul à la fois, et sous le contrôle de MSCS. En cas de reprise en ligne, MSCS bascule l'accès vers l'autre ordinateur. Le disque commun est généralement un disque RAID mais pas obligatoirement.

Chaque ordinateur est connecté au réseau externe pour accès client et chacun dispose d'une adresse IP. Toutefois, un client externe, qui communique avec ce cluster, ne connaît qu'une seule *adresse IP virtuelle* et MSCS achemine correctement le trafic IP au sein du cluster.

MSCS effectue également ses propres communications entre les deux ordinateurs, soit via une ou plusieurs connexions privées, soit via le réseau public, par exemple pour surveiller leurs états à l'aide du signal de présence et pour synchroniser leurs bases de données.

Windows Configuration de IBM MQ pour la mise en cluster MSCS

Vous configurez IBM MQ pour la mise en cluster en faisant du gestionnaire de files d'attente l'unité de basculement sur MSCS. Vous définissez un gestionnaire de files d'attente en tant que ressource sur MSCS, qui peut ensuite le surveiller, puis le transférer sur un autre ordinateur du cluster en cas de problème.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

Pour configurer votre système pour cela, commencez par installer IBM MQ sur chaque ordinateur du cluster.

Comme le gestionnaire de files d'attente est associé au nom d'installation IBM MQ, le nom d'installation IBM MQ sur tous les ordinateurs du cluster doit être identique. Voir [Installation et désinstallation](#).

Les gestionnaires de files d'attente eux-mêmes doivent exister uniquement sur l'ordinateur sur lequel vous les créez. En cas de reprise en ligne, MSCS lance les gestionnaires de files d'attente sur l'autre ordinateur. Toutefois, les gestionnaires de files d'attente doivent disposer de leurs fichiers journaux et de leurs fichiers de données sur un disque partagé de cluster et non sur une unité locale. Si un gestionnaire de files d'attente est déjà installé sur une unité locale, vous pouvez le migrer à l'aide d'un outil fourni avec IBM MQ; voir «[Transfert d'un gestionnaire de files d'attente vers le stockage MSCS](#)», à la page 533. Si vous souhaitez créer des gestionnaires de files d'attente à utiliser avec MSCS, voir «[Création d'un gestionnaire de files d'attente à utiliser avec MSCS](#)», à la page 532.

Après l'installation et la migration, utilisez l'administrateur de cluster MSCS pour informer MSCS de vos gestionnaires de files d'attente ; voir «[Placement d'un gestionnaire de files d'attente sous contrôle MSCS](#)», à la page 535.

Si vous décidez de supprimer un gestionnaire de files d'attente du contrôle MSCS, utilisez la procédure décrite dans «[Suppression d'un gestionnaire de files d'attente du contrôle MSCS](#)», à la page 542.

Windows *Symétrie de configuration et MSCS*

Lorsqu'une application passe d'un noeud à l'autre, elle doit se comporter de la même manière, quel que soit le noeud. La meilleure façon de s'en assurer est de rendre les environnements identiques.

Si vous le pouvez, configurez un cluster avec un matériel, un logiciel de système d'exploitation, un logiciel de produit et une configuration identiques sur chaque ordinateur. En particulier, assurez-vous que tous les logiciels requis installés sur les deux ordinateurs sont identiques en termes de version, de niveau de maintenance, de SupportPacs, de chemins et d'exits, et qu'il existe un espace de nom commun (environnement de sécurité) comme décrit dans «[Sécurité MSCS](#)», à la page 531.

Windows *Sécurité MSCS*

Pour une sécurité MSCS réussie, suivez ces instructions.

Les lignes directrices sont les suivantes:

- Vérifiez que vous disposez d'installations de logiciels identiques sur chaque ordinateur du cluster.
- Créez un espace de nom commun (environnement de sécurité) dans le cluster.
- Faites en sorte que les noeuds du cluster MSCS soient membres d'un domaine dans lequel le compte utilisateur qui est le *propriétaire du cluster* est un compte de domaine.
- Faites en sorte que les autres comptes utilisateur du cluster soient également des comptes de domaine, de sorte qu'ils soient disponibles sur les deux noeuds. C'est automatiquement le cas si vous disposez déjà d'un domaine et que les comptes associés à IBM MQ sont des comptes de domaine. Si vous ne disposez pas actuellement d'un domaine, envisagez de configurer un *mini-domaine* pour répondre aux besoins des noeuds de cluster et des comptes appropriés. Votre objectif est de faire en sorte que votre cluster de deux ordinateurs ressemble à une seule ressource informatique.

N'oubliez pas qu'un compte local sur un ordinateur n'existe pas sur l'autre. Même si vous créez un compte avec le même nom sur l'autre ordinateur, son identificateur de sécurité (SID) est différent, de sorte que lorsque votre application est déplacée vers l'autre noeud, les droits n'existent pas sur ce noeud.

Lors d'un basculement ou d'un déplacement, la prise en charge de IBM MQ MSCS garantit que tous les fichiers contenant des objets de gestionnaire de files d'attente disposent de droits équivalents sur le noeud de destination. Explicitement, le code vérifie que les administrateurs et les groupes mqm, ainsi que le compte SYSTEM, ont le contrôle total et que si Everyone avait un accès en lecture sur l'ancien noeud, ce droit est ajouté sur le noeud de destination.

Vous pouvez utiliser un compte de domaine pour exécuter votre service IBM MQ . Assurez-vous qu'il existe dans le groupe mqm local sur chaque ordinateur du cluster.

Windows *Utilisation de plusieurs gestionnaires de files d'attente avec MSCS*

Si vous exécutez plusieurs gestionnaires de files d'attente sur un ordinateur, vous pouvez choisir l'une de ces configurations.

Les configurations sont les suivantes:

- Tous les gestionnaires de files d'attente d'un même groupe. Dans cette configuration, si un problème se produit avec un gestionnaire de files d'attente, tous les gestionnaires de files d'attente du groupe basculant sur l'autre ordinateur en tant que groupe.
- Un seul gestionnaire de files d'attente dans chaque groupe. Dans cette configuration, si un problème se produit avec le gestionnaire de files d'attente, il est seul à effectuer une reprise sur l'autre ordinateur sans affecter les autres gestionnaires de files d'attente.
- Un mélange des deux premières configurations.

Windows Modes de cluster et MSCS

Il existe deux modes dans lesquels vous pouvez exécuter un système de cluster avec IBM MQ sous Windows: Actif / Passif ou Actif / Actif.

Remarque : Si vous utilisez MSCS avec Microsoft Transaction Server (COM +), vous ne pouvez pas utiliser le mode Actif / Actif.

Mode actif / passif

En mode actif / passif, l'ordinateur A est doté de l'application en cours d'exécution, et l'ordinateur B est de secours, utilisé uniquement lorsque MSCS détecte un problème.

Vous pouvez utiliser ce mode avec un seul disque partagé, mais si une application entraîne une reprise en ligne, **toutes** les applications doivent être transférées en tant que groupe (car un seul ordinateur peut accéder au disque partagé à la fois).

Vous pouvez configurer MSCS avec A comme ordinateur *préféré*. Ensuite, lorsque l'ordinateur A a été réparé ou remplacé et qu'il fonctionne à nouveau correctement, MSCS le détecte et rebascule automatiquement l'application sur l'ordinateur A.

Si vous exécutez plusieurs gestionnaires de files d'attente, envisagez d'avoir un disque partagé distinct pour chacun d'eux. Placez ensuite chaque gestionnaire de files d'attente dans un groupe distinct dans MSCS. De cette manière, tout gestionnaire de files d'attente peut basculer sur l'autre ordinateur sans affecter les autres gestionnaires de files d'attente.

Mode actif/actif

En mode actif / actif, les ordinateurs A et B ont tous deux des applications en cours d'exécution et les groupes de chaque ordinateur sont définis pour utiliser l'autre ordinateur comme sauvegarde. Si un incident est détecté sur l'ordinateur A, MSCS transfère les données d'état à l'ordinateur B et y lance à nouveau l'application. L'ordinateur B exécute alors sa propre application et celle de A.

Pour cette configuration, vous avez besoin d'au moins deux disques partagés. Vous pouvez configurer MSCS avec A comme ordinateur préféré pour les applications de A et B comme ordinateur préféré pour les applications de B. Après la reprise en ligne et la réparation, chaque application se retrouve automatiquement sur son propre ordinateur.

Pour IBM MQ, cela signifie que vous pouvez, par exemple, exécuter deux gestionnaires de files d'attente, un sur chacun des A et B, chacun exploitant toute la puissance de son propre ordinateur. Après un incident sur l'ordinateur A, les deux gestionnaires de files d'attente s'exécutent sur l'ordinateur B. Cela implique de partager la puissance d'un seul ordinateur, avec une capacité réduite à traiter de grandes quantités de données à la vitesse. Cependant, vos applications critiques seront toujours disponibles pendant que vous trouverez et réparez la panne sur A.

Windows ***Création d'un gestionnaire de files d'attente à utiliser avec MSCS***

Cette procédure permet de s'assurer qu'un nouveau gestionnaire de files d'attente est créé de manière à pouvoir être préparé et placé sous le contrôle de Microsoft Cluster Service (MSCS).

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

Vous commencez par créer le gestionnaire de files d'attente avec toutes ses ressources sur une unité locale, puis vous migrez les fichiers journaux et les fichiers de données sur un disque partagé. (Vous pouvez inverser cette opération.) N'essayez **pas** de créer un gestionnaire de files d'attente avec ses ressources sur une unité partagée.

Vous pouvez créer un gestionnaire de files d'attente à utiliser avec MSCS de deux manières, à partir d'une invite de commande ou dans IBM MQ Explorer. L'avantage de l'utilisation d'une invite de commande est que le gestionnaire de files d'attente est créé *arrêté* et défini sur le *démarrage manuel*, qui est prêt pour MSCS. (Le IBM MQ Explorer démarre automatiquement un nouveau gestionnaire de files d'attente et le définit sur le démarrage automatique après sa création. Vous devez changer cela.)

Création d'un gestionnaire de files d'attente à partir d'une invite de commande

Pour créer un gestionnaire de files d'attente à partir d'une invite de commande à utiliser avec MSCS, procédez comme suit:

1. Vérifiez que la variable d'environnement MQSPREFIX est définie pour faire référence à une unité locale, par exemple C:\IBM\MQ. Si vous modifiez cela, réamorçez la machine pour que le compte système récupère la modification. Si vous ne définissez pas la variable, le gestionnaire de files d'attente est créé dans le répertoire par défaut IBM MQ pour les gestionnaires de files d'attente.
2. Créez le gestionnaire de files d'attente à l'aide de la commande **crtmqm**. Par exemple, pour créer un gestionnaire de files d'attente appelé `mcs_test` dans le répertoire par défaut, utilisez:

```
crtmqm mcs_test
```

3. Passez à la section [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.

Création d'un gestionnaire de files d'attente à l'aide de IBM MQ Explorer

Pour créer un gestionnaire de files d'attente à l'aide de IBM MQ Explorer, à utiliser avec MSCS, procédez comme suit:

1. Démarrez le IBM MQ Explorer à partir du menu Démarrer.
2. Dans la vue Navigator, développez les noeuds d'arborescence pour rechercher le noeud d'arborescence `Gestionnaires de files d'attente`.
3. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence `Gestionnaires de files d'attente` et sélectionnez **Nouveau > Gestionnaire de files d'attente**. Le panneau Créer un gestionnaire de files d'attente s'affiche.
4. Complétez la boîte de dialogue (étape 1), puis cliquez sur **Suivant >**.
5. Complétez la boîte de dialogue (étape 2), puis cliquez sur **Suivant >**.
6. Complétez la boîte de dialogue (étape 3), en vous assurant que `Démarrer le gestionnaire de files d'attente` et `Créer un canal de connexion serveur` ne sont pas sélectionnés, puis cliquez sur **Suivant >**.
7. Complétez la boîte de dialogue (étape 4), puis cliquez sur **Terminer**.
8. Passez à la section [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.

Transfert d'un gestionnaire de files d'attente vers le stockage MSCS

Cette procédure configure un gestionnaire de files d'attente existant pour qu'il puisse être placé sous le contrôle de Microsoft Cluster Service (MSCS).

Pour ce faire, vous déplacez les fichiers journaux et les fichiers de données sur des disques partagés afin de les mettre à la disposition de l'autre ordinateur en cas de défaillance. Par exemple, le gestionnaire de files d'attente existant peut avoir des chemins tels que `C:\WebSphere\MQ\log\QMname` et `C:\WebSphere\MQ\qmgrs\QMname`.



Avvertissement : N'essayez pas de déplacer les fichiers manuellement ; utilisez le programme utilitaire fourni dans le cadre du support IBM MQ MSCS, comme décrit dans cette rubrique.

Si le gestionnaire de files d'attente en cours de déplacement utilise des connexions TLS et que le référentiel de clés TLS se trouve dans le répertoire de données du gestionnaire de files d'attente sur la machine locale, le référentiel de clés est déplacé avec le reste du gestionnaire de files d'attente vers le disque partagé. Par défaut, l'attribut de gestionnaire de files d'attente qui spécifie l'emplacement du référentiel de clés TLS, `SSLKEYR`, est défini sur `MQ_INSTALLATION_PATH\qmgrs\QMGRNAME\ssl\key`, qui se trouve sous le répertoire de données du gestionnaire de files d'attente. `MQ_INSTALLATION_PATH` représente le répertoire de haut niveau dans lequel IBM MQ est installé. La commande `hamvmqm` ne modifie pas cet attribut de gestionnaire de files d'attente. Dans cette situation, vous devez modifier l'attribut de gestionnaire de files d'attente, `SSLKEYR`, à l'aide de IBM MQ Explorer ou de la commande `MQSC ALTER QMGR`, pour qu'il pointe vers le nouveau fichier de référentiel de clés TLS.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

La procédure est la suivante:

1. Arrêtez le gestionnaire de files d'attente et vérifiez qu'il n'y a pas d'erreurs.
2. Si les fichiers journaux ou les fichiers de file d'attente du gestionnaire de files d'attente sont déjà stockés sur un disque partagé, ignorez le reste de cette procédure et passez directement à l'étape «Placement d'un gestionnaire de files d'attente sous contrôle MSCS», à la page 535.
3. Effectuez une sauvegarde de support intégrale des fichiers de file d'attente et des fichiers journaux et stockez la sauvegarde dans un endroit sûr (voir «Fichiers journaux du gestionnaire de files d'attente», à la page 545 pour savoir pourquoi cela est important).
4. Si vous disposez déjà d'une ressource de disque partagé appropriée, passez à l'étape 6. Sinon, utilisez l'administrateur de cluster MSCS pour créer une ressource de type *disque partagé* avec une capacité suffisante pour stocker les fichiers journaux du gestionnaire de files d'attente et les fichiers de données (file d'attente).
5. Testez le disque partagé à l'aide de l'administrateur de cluster MSCS pour le déplacer d'un noeud de cluster à l'autre, puis de nouveau.
6. Assurez-vous que le disque partagé est en ligne sur le noeud de cluster où le journal du gestionnaire de files d'attente et les fichiers de données sont stockés localement.
7. Exécutez le programme utilitaire pour déplacer le gestionnaire de files d'attente comme suit:

```
hamvmqm /m qmname /dd " e: \  
IBM MQ " /ld " e: \  
IBM MQ \log"
```

en remplaçant le nom de votre gestionnaire de files d'attente par `qmname`, votre identificateur d'unité de disque partagé pour `eet` le répertoire que vous avez choisi pour `IBM MQ`. Les répertoires sont créés s'ils n'existent pas déjà.

8. Testez le gestionnaire de files d'attente pour vous assurer qu'il fonctionne à l'aide de IBM MQ Explorer. Exemple :
 - a. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence du gestionnaire de files d'attente, puis sélectionnez **Démarrer**. Le gestionnaire de files d'attente démarre.
 - b. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence *Files d'attente*, puis sélectionnez **Nouveau > File d'attente locale ...**, et attribuez un nom à la file d'attente.
 - c. Cliquez sur **Terminer**.
 - d. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Insertion d'un message de test ...**. Le panneau Insertion d'un message de test s'affiche.
 - e. Entrez un texte de message, puis cliquez sur **Insertion d'un message de test** et fermez le panneau.

- f. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Parcourir les messages** Le panneau Navigateur de messages s'affiche.
 - g. Vérifiez que votre message se trouve dans la file d'attente, puis cliquez sur **Fermer**. Le panneau Navigateur de messages se ferme.
 - h. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Effacer les messages** Les messages de la file d'attente sont effacés.
 - i. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Supprimer** Un panneau de confirmation s'affiche. Cliquez sur **OK**. La file d'attente est supprimée.
 - j. Cliquez avec le bouton droit de la souris sur le noeud de l'arborescence du gestionnaire de files d'attente, puis sélectionnez **Arrêter** Le panneau Arrêt du gestionnaire de files d'attente s'affiche.
 - k. Cliquez sur **OK**. Le gestionnaire de files d'attente s'arrête.
9. En tant qu'administrateur IBM MQ , vérifiez que l'attribut de démarrage du gestionnaire de files d'attente est défini sur manuel. Dans IBM MQ Explorer, définissez la zone Démarrage sur manuel dans le panneau des propriétés du gestionnaire de files d'attente.
 10. Passez à la section [«Placement d'un gestionnaire de files d'attente sous contrôle MSCS»](#), à la page 535.

Windows *Placement d'un gestionnaire de files d'attente sous contrôle MSCS*

Comment placer un gestionnaire de files d'attente sous le contrôle Microsoft Cluster Service (MSCS), y compris les tâches prérequis.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

Avant de placer un gestionnaire de files d'attente sous le contrôle MSCS/WSFC

Avant de placer un gestionnaire de files d'attente sous le contrôle MSCS/WSFC, procédez comme suit:

1. Vérifiez que IBM MQ et son support MSCS/WSFC sont installés sur les deux machines du cluster et que les logiciels de chaque ordinateur sont identiques, comme décrit dans [«Configuration de IBM MQ pour la mise en cluster MSCS»](#), à la page 530.
2. Utilisez le programme utilitaire **haretyp** pour enregistrer IBM MQ en tant que type de ressource MSCS sur tous les noeuds de cluster. Voir [«Prise en charge des programmes utilitaires MSCS»](#), à la page 546.
3. Si vous ne l'avez pas encore fait, [créez un gestionnaire de files d'attente à utiliser avec MSCS/WSFC](#).
4. Si vous avez créé le gestionnaire de files d'attente ou s'il existe déjà, vérifiez que vous avez effectué la procédure décrite dans [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.
5. Si le gestionnaire de files d'attente est en cours d'exécution, arrêtez-le à l'aide d'une invite de commande ou de l'explorateur IBM MQ .
6. Testez le fonctionnement MSCS/WSFC des unités partagées avant de passer à l'une des procédures Windows suivantes dans cette rubrique.

Windows Server 2012, 2016, 2019 ou 2022

Pour placer un gestionnaire de files d'attente sous le contrôle MSCS/WSFC sur Windows Server 2012 ou version ultérieure, procédez comme suit:

1. Connectez-vous à l'ordinateur du noeud de cluster hébergeant le gestionnaire de files d'attente ou connectez-vous à un poste de travail distant en tant qu'utilisateur disposant de droits d'administration de cluster, puis connectez-vous au noeud de cluster hébergeant le gestionnaire de files d'attente.
2. Démarrez l'outil de gestion du cluster de basculement.

3. Cliquez avec le bouton droit de la souris sur **Failover Cluster Management > Connect Cluster ...** pour ouvrir une connexion au cluster.
4. Contrairement au schéma de groupe utilisé dans MSCS Cluster Administrator sur les versions précédentes de Windows, l'outil Failover Cluster Management utilise le concept de services et d'applications. Un service ou une application configurée contient toutes les ressources nécessaires pour qu'une application soit mise en cluster. Vous pouvez configurer un gestionnaire de files d'attente sous WSFC comme suit:
 - a. Cliquez avec le bouton droit de la souris sur le cluster et sélectionnez **Configurer le rôle** pour démarrer l'assistant de configuration.
 - b. Sélectionnez **Autre serveur** dans le panneau **Sélectionner un service ou une application**.
 - c. Sélectionnez une adresse IP appropriée comme point d'accès client.
 Cette adresse doit être une adresse IP inutilisée que les clients et les autres gestionnaires de files d'attente doivent utiliser pour se connecter au gestionnaire de files d'attente *virtuel*. Cette adresse IP n'est pas l'adresse normale (statique) de l'un des noeuds ; il s'agit d'une adresse supplémentaire qui *flotte* entre eux. Bien que WSFC gère le routage de cette adresse, elle ne vérifie **pas** que l'adresse est accessible.
 - d. Affectez une unité de stockage pour une utilisation exclusive par le gestionnaire de files d'attente. Ce périphérique doit être créé en tant qu'instance de ressource avant de pouvoir être affecté.
 Vous pouvez utiliser une unité pour stocker à la fois les journaux et les fichiers de file d'attente, ou vous pouvez les répartir sur plusieurs unités. Dans les deux cas, si chaque gestionnaire de files d'attente possède son propre disque partagé, assurez-vous que toutes les unités utilisées par ce gestionnaire de files d'attente sont exclusives à ce gestionnaire de files d'attente, c'est-à-dire que rien d'autre ne repose sur les unités. Veillez également à créer une instance de ressource pour chaque unité utilisée par le gestionnaire de files d'attente.
 Le type de ressource d'une unité dépend de la prise en charge SCSI que vous utilisez ; reportez-vous aux instructions de votre adaptateur SCSI. Il se peut qu'il existe déjà des groupes et des ressources pour chacune des unités partagées. Si tel est le cas, vous n'avez pas besoin de créer l'instance de ressource pour chaque unité. Déplacez-la de son groupe en cours vers celui créé pour le gestionnaire de files d'attente.
 Pour chaque ressource d'unité, définissez les propriétaires possibles sur les deux noeuds. Définissez les ressources dépendantes sur aucune.
 - e. Sélectionnez la ressource **MQSeries MSCS** dans le panneau **Sélectionner le type de ressource**.
 - f. Effectuez les étapes restantes de l'assistant.
5. Avant de mettre la ressource en ligne, la ressource MQSeries MSCS nécessite une configuration supplémentaire:
 - a. Sélectionnez le service nouvellement défini qui contient une ressource appelée 'New MQSeries MSCS'.
 - b. Cliquez avec le bouton droit de la souris sur **Propriétés** dans la ressource IBM MQ.
 - c. Configurez la ressource:
 - Name ; choisissez un nom qui facilite l'identification du gestionnaire de files d'attente auquel il est destiné.
 - Run in a separate Resource Monitor ; pour un meilleur isolement
 - Possible owners ; définir les deux noeuds
 - Dependencies ; ajouter l'unité et l'adresse IP de ce gestionnaire de files d'attente.

Avertissement : Si vous n'ajoutez pas ces dépendances, IBM MQ tente d'écrire le statut du gestionnaire de files d'attente sur le mauvais disque de cluster lors des reprises en ligne. Etant donné que de nombreux processus peuvent tenter d'écrire simultanément sur ce disque, certains processus IBM MQ peuvent être bloqués.

 - Parameters ; comme suit:

- QueueManagerName (obligatoire) ; nom du gestionnaire de files d'attente que cette ressource doit contrôler. Ce gestionnaire de files d'attente doit exister sur l'ordinateur local.
- PostOnlineCommand (facultatif) ; vous pouvez spécifier un programme à exécuter chaque fois que la ressource du gestionnaire de files d'attente passe de l'état hors ligne à l'état en ligne. Pour plus de détails, voir [«Commande PostOnlineet commande PreOfflinedans MSCS»](#), à la page 545.
- PreOfflineCommand (facultatif) ; vous pouvez spécifier un programme à exécuter chaque fois que la ressource du gestionnaire de files d'attente passe de l'état en ligne à l'état hors ligne. Pour plus de détails, voir [«Commande PostOnlineet commande PreOfflinedans MSCS»](#), à la page 545.

Remarque : L'intervalle d'interrogation *looksAlive* est défini sur la valeur par défaut de 5000 ms. L'intervalle d'interrogation *isAlive* est défini sur la valeur par défaut de 60000 ms. Ces valeurs par défaut ne peuvent être modifiées qu'une fois la définition de ressource terminée. Pour plus de détails, voir [«Interrogation looksAlive et isAlive sur MSCS»](#), à la page 542.

- d. Vous pouvez éventuellement définir un noeud préférentiel (mais notez les commentaires dans [«Utilisation de noeuds favoris dans MSCS»](#), à la page 546)
 - e. La *règle de reprise en ligne* est définie par défaut sur des valeurs sensibles, mais vous pouvez optimiser les seuils et les périodes qui contrôlent la *reprise en ligne des ressources* et la *reprise en ligne du groupe* pour qu'elles correspondent aux charges placées sur le gestionnaire de files d'attente.
6. Testez le gestionnaire de files d'attente en le mettant en ligne dans l'administrateur de cluster MSCS et en le soumettant à une charge de travail de test. Si vous testez un gestionnaire de files d'attente de test, utilisez l'explorateur IBM MQ . Exemple :
 - a. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence Files d'attente , puis sélectionnez **Nouveau > File d'attente locale ...**, et attribuez un nom à la file d'attente.
 - b. Cliquez sur **Terminer**. La file d'attente est créée et affichée dans la vue de contenu.
 - c. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Insertion d'un message de test ...**. Le panneau Insertion d'un message de test s'affiche.
 - d. Entrez un texte de message, puis cliquez sur **Insertion d'un message de test** et fermez le panneau.
 - e. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Parcourir les messages ...**. Le panneau Navigateur de messages s'affiche.
 - f. Vérifiez que votre message se trouve dans la file d'attente, puis cliquez sur **Fermer**. Le panneau Navigateur de messages se ferme.
 - g. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Effacer les messages ...**. Les messages de la file d'attente sont effacés.
 - h. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Supprimer ...**. Un panneau de confirmation s'affiche. Cliquez sur **OK**. La file d'attente est supprimée.
 7. Vérifiez que le gestionnaire de files d'attente peut être mis hors ligne et de nouveau en ligne à l'aide de l'administrateur de cluster MSCS.
 8. Simulez une reprise en ligne.

Dans l'administrateur de cluster MSCS, cliquez avec le bouton droit de la souris sur le groupe contenant le gestionnaire de files d'attente et sélectionnez **Move Group**. Cette opération peut prendre quelques minutes. (Si, à d'autres moments, vous souhaitez déplacer rapidement un gestionnaire de files d'attente vers un autre noeud, suivez la procédure décrite dans [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.) Vous pouvez également cliquer avec le bouton droit de la souris et sélectionner **Initiate Failure** ; l'action (redémarrage local ou reprise en ligne) dépend de l'état en cours et des paramètres de configuration.

Windows Server 2008

Pour placer un gestionnaire de files d'attente sous le contrôle de MSCS sur Windows Server 2008, procédez comme suit:

1. Connectez-vous à l'ordinateur du noeud de cluster hébergeant le gestionnaire de files d'attente ou connectez-vous à un poste de travail distant en tant qu'utilisateur disposant de droits d'administration de cluster, puis connectez-vous au noeud de cluster hébergeant le gestionnaire de files d'attente.
2. Démarrez l'outil de gestion du cluster de basculement.
3. Cliquez avec le bouton droit de la souris sur **Failover Cluster Management > Manage a Cluster ...** pour ouvrir une connexion au cluster.
4. Contrairement au schéma de groupe utilisé dans MSCS Cluster Administrator sur les versions précédentes de Windows, l'outil Failover Cluster Management utilise le concept de services et d'applications. Un service ou une application configurée contient toutes les ressources nécessaires pour qu'une application soit mise en cluster. Vous pouvez configurer un gestionnaire de files d'attente sous MSCS comme suit:
 - a. Cliquez avec le bouton droit de la souris sur **Services et applications > Configurer un service ou une application ...** pour démarrer l'assistant de configuration.
 - b. Sélectionnez **Autre serveur** dans le panneau **Sélectionner un service ou une application**.
 - c. Sélectionnez une adresse IP appropriée comme point d'accès client.

Cette adresse doit être une adresse IP inutilisée que les clients et les autres gestionnaires de files d'attente doivent utiliser pour se connecter au gestionnaire de files d'attente *virtuel*. Cette adresse IP n'est pas l'adresse normale (statique) de l'un des noeuds ; il s'agit d'une adresse supplémentaire qui *flotte* entre eux. Bien que MSCS gère le routage de cette adresse, elle ne vérifie **pas** que l'adresse est accessible.

- d. Affectez une unité de stockage pour une utilisation exclusive par le gestionnaire de files d'attente. Ce périphérique doit être créé en tant qu'instance de ressource avant de pouvoir être affecté.

Vous pouvez utiliser une unité pour stocker à la fois les journaux et les fichiers de file d'attente, ou vous pouvez les répartir sur plusieurs unités. Dans les deux cas, si chaque gestionnaire de files d'attente possède son propre disque partagé, assurez-vous que toutes les unités utilisées par ce gestionnaire de files d'attente sont exclusives à ce gestionnaire de files d'attente, c'est-à-dire que rien d'autre ne repose sur les unités. Veillez également à créer une instance de ressource pour chaque unité utilisée par le gestionnaire de files d'attente.

Le type de ressource d'une unité dépend de la prise en charge SCSI que vous utilisez ; reportez-vous aux instructions de votre adaptateur SCSI. Il se peut qu'il existe déjà des groupes et des ressources pour chacune des unités partagées. Si tel est le cas, vous n'avez pas besoin de créer l'instance de ressource pour chaque unité. Déplacez-la de son groupe en cours vers celui créé pour le gestionnaire de files d'attente.

Pour chaque ressource d'unité, définissez les propriétaires possibles sur les deux noeuds. Définissez les ressources dépendantes sur aucune.

- e. Sélectionnez la ressource **MQSeries MSCS** dans le panneau **Sélectionner le type de ressource**.
 - f. Effectuez les étapes restantes de l'assistant.
5. Avant de mettre la ressource en ligne, la ressource MQSeries MSCS nécessite une configuration supplémentaire:
 - a. Sélectionnez le service nouvellement défini qui contient une ressource appelée 'New MQSeries MSCS'.
 - b. Cliquez avec le bouton droit de la souris sur **Propriétés** sur la ressource MQ.
 - c. Configurez la ressource:
 - Name ; choisissez un nom qui facilite l'identification du gestionnaire de files d'attente auquel il est destiné.
 - Run in a separate Resource Monitor ; pour un meilleur isolement

- Possible owners ; définir les deux noeuds
- Dependencies ; ajouter l'unité et l'adresse IP de ce gestionnaire de files d'attente.

Avertissement : Si vous n'ajoutez pas ces dépendances, IBM MQ tente d'écrire le statut du gestionnaire de files d'attente sur le mauvais disque de cluster lors des reprises en ligne. Etant donné que de nombreux processus peuvent tenter d'écrire simultanément sur ce disque, certains processus IBM MQ peuvent être bloqués.

- Parameters ; comme suit:
 - QueueManagerName (obligatoire) ; nom du gestionnaire de files d'attente que cette ressource doit contrôler. Ce gestionnaire de files d'attente doit exister sur l'ordinateur local.
 - PostOnlineCommand (facultatif) ; vous pouvez spécifier un programme à exécuter chaque fois que la ressource du gestionnaire de files d'attente passe de l'état hors ligne à l'état en ligne. Pour plus de détails, voir [«Commande PostOnline et commande PreOffline dans MSCS»](#), à la page 545.
 - PreOfflineCommand (facultatif) ; vous pouvez spécifier un programme à exécuter chaque fois que la ressource du gestionnaire de files d'attente passe de l'état en ligne à l'état hors ligne. Pour plus de détails, voir [«Commande PostOnline et commande PreOffline dans MSCS»](#), à la page 545.

Remarque : L'intervalle d'interrogation *looksAlive* est défini sur la valeur par défaut de 5000 ms. L'intervalle d'interrogation *isAlive* est défini sur la valeur par défaut de 60000 ms. Ces valeurs par défaut ne peuvent être modifiées qu'une fois la définition de ressource terminée. Pour plus de détails, voir [«Interrogation looksAlive et isAlive sur MSCS»](#), à la page 542.

- d. Vous pouvez éventuellement définir un noeud préférentiel (mais notez les commentaires dans [«Utilisation de noeuds favoris dans MSCS»](#), à la page 546)
 - e. La *règle de reprise en ligne* est définie par défaut sur des valeurs sensibles, mais vous pouvez optimiser les seuils et les périodes qui contrôlent la *reprise en ligne des ressources* et la *reprise en ligne du groupe* pour qu'elles correspondent aux charges placées sur le gestionnaire de files d'attente.
6. Testez le gestionnaire de files d'attente en le mettant en ligne dans l'administrateur de cluster MSCS et en le soumettant à une charge de travail de test. Si vous testez un gestionnaire de files d'attente de test, utilisez l'explorateur IBM MQ . Exemple :
 - a. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence Files d'attente , puis sélectionnez **Nouveau > File d'attente locale ...**, et attribuez un nom à la file d'attente.
 - b. Cliquez sur **Terminer**. La file d'attente est créée et affichée dans la vue de contenu.
 - c. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Insertion d'un message de test ...**. Le panneau **Insertion d'un message de test** s'affiche.
 - d. Entrez un texte de message, puis cliquez sur **Insertion d'un message de test** et fermez le panneau.
 - e. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Parcourir les messages ...**. Le panneau **Navigateur de messages** s'affiche.
 - f. Vérifiez que votre message se trouve dans la file d'attente, puis cliquez sur **Fermer**. Le panneau **Navigateur de messages** se ferme.
 - g. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Effacer les messages ...**. Les messages de la file d'attente sont effacés.
 - h. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Supprimer ...**. Un panneau de confirmation s'affiche. Cliquez sur **OK**. La file d'attente est supprimée.
 7. Vérifiez que le gestionnaire de files d'attente peut être mis hors ligne et de nouveau en ligne à l'aide de l'administrateur de cluster MSCS.
 8. Simulez une reprise en ligne.

Dans l'administrateur de cluster MSCS, cliquez avec le bouton droit de la souris sur le groupe contenant le gestionnaire de files d'attente et sélectionnez **Move Group**. Cette opération peut prendre quelques minutes. (Si, à d'autres moments, vous souhaitez déplacer rapidement un gestionnaire de

files d'attente vers un autre noeud, suivez la procédure décrite dans «[Transfert d'un gestionnaire de files d'attente vers le stockage MSCS](#)», à la page 533.) Vous pouvez également cliquer avec le bouton droit de la souris et sélectionner **Initiate Failure** ; l'action (redémarrage local ou reprise en ligne) dépend de l'état en cours et des paramètres de configuration.

Windows 2003

Pour placer un gestionnaire de files d'attente sous le contrôle MSCS sous Windows 2003, procédez comme suit:

1. Connectez-vous à l'ordinateur du noeud de cluster hébergeant le gestionnaire de files d'attente ou connectez-vous à un poste de travail distant en tant qu'utilisateur disposant de droits d'administration de cluster, puis connectez-vous au noeud de cluster hébergeant le gestionnaire de files d'attente.
2. Démarrez l'administrateur de cluster MSCS.
3. Ouvrez une connexion au cluster.
4. Créez un groupe MSCS à utiliser pour contenir les ressources du gestionnaire de files d'attente. Nommez le groupe de telle sorte qu'il soit évident à quel gestionnaire de files d'attente il est lié. Chaque groupe peut contenir plusieurs gestionnaires de files d'attente, comme décrit dans «[Utilisation de plusieurs gestionnaires de files d'attente avec MSCS](#)», à la page 531.

Utilisez le groupe pour toutes les étapes restantes.

5. Créez une instance de ressource pour chacune des unités logiques SCSI utilisées par le gestionnaire de files d'attente.

Vous pouvez utiliser une unité pour stocker à la fois les journaux et les fichiers de file d'attente, ou vous pouvez les répartir sur plusieurs unités. Dans les deux cas, si chaque gestionnaire de files d'attente possède son propre disque partagé, assurez-vous que toutes les unités utilisées par ce gestionnaire de files d'attente sont exclusives à ce gestionnaire de files d'attente, c'est-à-dire que rien d'autre ne repose sur les unités. Veillez également à créer une instance de ressource pour chaque unité utilisée par le gestionnaire de files d'attente.

Le type de ressource d'une unité dépend de la prise en charge SCSI que vous utilisez ; reportez-vous aux instructions de votre adaptateur SCSI. Il se peut qu'il existe déjà des groupes et des ressources pour chacune des unités partagées. Si tel est le cas, vous n'avez pas besoin de créer l'instance de ressource pour chaque unité. Déplacez-la de son groupe en cours vers celui créé pour le gestionnaire de files d'attente.

Pour chaque ressource d'unité, définissez les propriétaires possibles sur les deux noeuds. Définissez les ressources dépendantes sur aucune.

6. Créez une instance de ressource pour l'adresse IP.

Créez une ressource d'adresse IP (type de ressource *Adresse IP*). Cette adresse doit être une adresse IP inutilisée que les clients et les autres gestionnaires de files d'attente doivent utiliser pour se connecter au gestionnaire de files d'attente *virtuel*. Cette adresse IP n'est pas l'adresse normale (statique) de l'un des noeuds ; il s'agit d'une adresse supplémentaire qui *flotte* entre eux. Bien que MSCS gère le routage de cette adresse, elle ne vérifie **pas** que l'adresse est accessible.

7. Créez une instance de ressource pour le gestionnaire de files d'attente.

Créez une ressource de type *IBM MQ MSCS*. L'assistant vous invite à entrer divers éléments, notamment:

- **Name** ; choisissez un nom qui facilite l'identification du gestionnaire de files d'attente auquel il est destiné.
- **Add to group** ; utiliser le groupe que vous avez créé
- **Run in a separate Resource Monitor** ; pour un meilleur isolement
- **Possible owners** ; définir les deux noeuds
- **Dependencies** ; ajouter l'unité et l'adresse IP de ce gestionnaire de files d'attente.

Avertissement : Si vous n'ajoutez pas ces dépendances, IBM MQ tente d'écrire le statut du gestionnaire de files d'attente sur le mauvais disque de cluster lors des reprises en ligne. Etant donné que de nombreux processus peuvent tenter d'écrire simultanément sur ce disque, certains processus IBM MQ peuvent être bloqués.

- Parameters ; comme suit:
 - QueueManagerName (obligatoire) ; nom du gestionnaire de files d'attente que cette ressource doit contrôler. Ce gestionnaire de files d'attente doit exister sur l'ordinateur local.
 - PostOnlineCommand (facultatif) ; vous pouvez spécifier un programme à exécuter chaque fois que la ressource du gestionnaire de files d'attente passe de l'état hors ligne à l'état en ligne. Pour plus de détails, voir [«Commande PostOnlineet commande PreOfflinedans MSCS»](#), à la page 545.
 - PreOfflineCommand (facultatif) ; vous pouvez spécifier un programme à exécuter chaque fois que la ressource du gestionnaire de files d'attente passe de l'état en ligne à l'état hors ligne. Pour plus de détails, voir [«Commande PostOnlineet commande PreOfflinedans MSCS»](#), à la page 545.

Remarque : L'intervalle d'interrogation *looksAlive* est défini sur la valeur par défaut de 5000 ms. L'intervalle d'interrogation *isAlive* est défini sur la valeur par défaut de 30000 ms. Ces valeurs par défaut ne peuvent être modifiées qu'une fois la définition de ressource terminée. Pour plus de détails, voir [«Interrogation looksAlive et isAlive sur MSCS»](#), à la page 542.

8. Vous pouvez éventuellement définir un noeud préférentiel (mais notez les commentaires dans [«Utilisation de noeuds favoris dans MSCS»](#), à la page 546)
9. La *règle de reprise en ligne* (telle que définie dans les propriétés du groupe) est définie par défaut sur des valeurs sensibles, mais vous pouvez ajuster les seuils et les périodes qui contrôlent *Reprise en ligne des ressources* et *Reprise en ligne du groupe* pour qu'ils correspondent aux charges placées sur le gestionnaire de files d'attente.
10. Testez le gestionnaire de files d'attente en le mettant en ligne dans l'administrateur de cluster MSCS et en le soumettant à une charge de travail de test. Si vous testez un gestionnaire de files d'attente de test, utilisez l'explorateur IBM MQ . Exemple :
 - a. Cliquez avec le bouton droit de la souris sur le noeud d'arborescence Files d'attente , puis sélectionnez **Nouveau > File d'attente locale ...**, et attribuez un nom à la file d'attente.
 - b. Cliquez sur **Terminer**. La file d'attente est créée et affichée dans la vue de contenu.
 - c. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Insertion d'un message de test ...**. Le panneau **Insertion d'un message de test** s'affiche.
 - d. Entrez un texte de message, puis cliquez sur **Insertion d'un message de test** et fermez le panneau.
 - e. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Parcourir les messages ...**. Le panneau **Navigateur de messages** s'affiche.
 - f. Vérifiez que votre message se trouve dans la file d'attente, puis cliquez sur **Fermer**. Le panneau **Navigateur de messages** se ferme.
 - g. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Effacer les messages ...**. Les messages de la file d'attente sont effacés.
 - h. Cliquez avec le bouton droit de la souris sur la file d'attente, puis sélectionnez **Supprimer ...**. Un panneau de confirmation s'affiche. Cliquez sur **OK**. La file d'attente est supprimée.
11. Vérifiez que le gestionnaire de files d'attente peut être mis hors ligne et de nouveau en ligne à l'aide de l'administrateur de cluster MSCS.
12. Simulez une reprise en ligne.

Dans l'administrateur de cluster MSCS, cliquez avec le bouton droit de la souris sur le groupe contenant le gestionnaire de files d'attente et sélectionnez **Move Group**. Cette opération peut prendre quelques minutes. (Si, à d'autres moments, vous souhaitez déplacer rapidement un gestionnaire de files d'attente vers un autre noeud, suivez la procédure décrite dans [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.) Vous pouvez également

cliquer avec le bouton droit de la souris et sélectionner *Initiate Failure* ; l'action (redémarrage local ou reprise en ligne) dépend de l'état en cours et des paramètres de configuration.

Windows *Interrogation looksAlive et isAlive sur MSCS*

looksAlive et *isAlive* sont des intervalles à partir desquels Microsoft Cluster Service (MSCS) effectue des appels dans le code de bibliothèque fourni par les types de ressource et demande à la ressource d'effectuer des vérifications afin de déterminer l'état de travail d'elle-même. Cela détermine en fin de compte si MSCS tente de basculer sur la ressource.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

Chaque fois que l'intervalle *looksAlive* s'écoule (5000 ms par défaut), la ressource du gestionnaire de files d'attente est appelée pour effectuer sa propre vérification afin de déterminer si son statut est satisfaisant.

Chaque fois que l'intervalle *isAlive* s'écoule (30000 ms par défaut), un autre appel est effectué à la ressource du gestionnaire de files d'attente pour qu'elle effectue une autre vérification afin de déterminer si la ressource fonctionne correctement. Cela active deux niveaux de vérification du type de ressource.

1. Une vérification de statut *looksAlive* pour déterminer si la ressource semble fonctionner.
2. Une vérification *isAlive* plus importante qui détermine si la ressource du gestionnaire de files d'attente est active.

Si la ressource du gestionnaire de files d'attente n'est pas active, MSCS, en fonction des autres options MSCS avancées, déclenche un basculement de la ressource et des ressources dépendantes associées vers un autre nœud du cluster. Pour plus d'informations, voir la [documentation MSCS](#).

Windows *Suppression d'un gestionnaire de files d'attente du contrôle MSCS*

Vous pouvez supprimer des gestionnaires de files d'attente du contrôle Microsoft Cluster Service (MSCS) et les renvoyer à l'administration manuelle.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

Il n'est pas nécessaire de supprimer les gestionnaires de files d'attente du contrôle MSCS pour les opérations de maintenance. Pour ce faire, vous pouvez mettre temporairement hors ligne un gestionnaire de files d'attente à l'aide de l'administrateur de cluster MSCS. La suppression d'un gestionnaire de files d'attente du contrôle MSCS est une modification plus permanente. Vous ne pouvez la supprimer que si vous décidez que MSCS ne doit plus avoir de contrôle supplémentaire sur le gestionnaire de files d'attente.

Si le gestionnaire de files d'attente est supprimé à l'aide de connexions TSL, vous devez modifier l'attribut de gestionnaire de files d'attente, *SSLKEYR*, à l'aide de l'IBM MQ Explorateur ou de la commande *MQSC ALTER QMGR*, pour qu'il pointe vers le fichier de référentiel de clés TLS sur le répertoire local.

La procédure est la suivante:

1. Mettez la ressource de gestionnaire de files d'attente hors ligne à l'aide de l'administrateur de cluster MSCS, comme décrit dans [«Mise hors ligne d'un gestionnaire de files d'attente à partir de MSCS»](#), à la page 542
2. Supprimez l'instance de ressource. Cela ne détruit pas le gestionnaire de files d'attente.
3. Le cas échéant, rémigrez les fichiers du gestionnaire de files d'attente des unités partagées vers les unités locales. Pour ce faire, voir [«Renvoi d'un gestionnaire de files d'attente à partir du stockage MSCS»](#), à la page 543.
4. Testez le gestionnaire de files d'attente.

Mise hors ligne d'un gestionnaire de files d'attente à partir de MSCS

Pour mettre un gestionnaire de files d'attente hors ligne à partir de MSCS, procédez comme suit:

1. Démarrez l'administrateur de cluster MSCS.

2. Ouvrez une connexion au cluster.
3. Sélectionnez `Groupsou Role` si vous utilisez Windows 2012 et ouvrez le groupe contenant le gestionnaire de files d'attente à déplacer.
4. Sélectionnez la ressource de gestionnaire de files d'attente.
5. Cliquez dessus avec le bouton droit de la souris et sélectionnez `Offline`.
6. Attendez la fin de l'opération.

Renvoi d'un gestionnaire de files d'attente à partir du stockage MSCS

Cette procédure configure le gestionnaire de files d'attente pour qu'il soit de nouveau sur l'unité locale de son ordinateur, c'est-à-dire qu'il devienne un gestionnaire de files d'attente *normal* IBM MQ. Pour ce faire, vous déplacez les fichiers journaux et les fichiers de données à partir des disques partagés. Par exemple, le gestionnaire de files d'attente existant peut avoir des chemins tels que `E:\WebSphere MQ\Log\QMname` et `E:\WebSphere MQ\qmgrs\QMname`. N'essayez pas de déplacer les fichiers manuellement ; utilisez le programme utilitaire **hamvmqm** fourni dans le cadre du support IBM MQ MSCS:

1. Effectuez une sauvegarde de support intégrale des fichiers de file d'attente et des fichiers journaux et stockez la sauvegarde dans un endroit sûr (voir «[Fichiers journaux du gestionnaire de files d'attente](#)», à la [page 545](#) pour savoir pourquoi cela est important).
2. Choisissez l'unité locale à utiliser et assurez-vous qu'elle dispose d'une capacité suffisante pour stocker les fichiers journaux du gestionnaire de files d'attente et les fichiers de données (file d'attente).
3. Assurez-vous que le disque partagé sur lequel se trouvent les fichiers est en ligne sur le noeud de cluster vers lequel déplacer les fichiers journaux et de données du gestionnaire de files d'attente.
4. Exécutez le programme utilitaire pour déplacer le gestionnaire de files d'attente comme suit:

```
hamvmqm /m qmname /dd " c:\
IBM MQ " /ld "c:\
IBM MQ \log"
```

en remplaçant le nom de votre gestionnaire de files d'attente par *qmname*, votre identificateur d'unité de disque local pour cet le répertoire que vous avez choisi pour *IBM MQ* (les répertoires sont créés s'ils n'existent pas déjà).

5. Testez le gestionnaire de files d'attente pour vous assurer qu'il fonctionne (comme décrit dans «[Transfert d'un gestionnaire de files d'attente vers le stockage MSCS](#)», à la [page 533](#)).

Windows *Conseils et astuces sur l'utilisation de MSCS*

Cette section contient des informations générales pour vous aider à utiliser efficacement le support IBM MQ pour Microsoft Cluster Service (MSCS).

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

Combien de temps faut-il pour faire échouer un gestionnaire de files d'attente d'une machine à l'autre? Cela dépend en grande partie de la quantité de charge de travail sur le gestionnaire de files d'attente et de la combinaison de trafic, par exemple, de la quantité de données persistantes dans le point de synchronisation et de la quantité de données validées avant l'échec. Les tests IBM ont donné des temps de reprise en ligne et de reprise par restauration d'environ une minute. Il s'agissait d'un gestionnaire de files d'attente très peu chargé et les temps réels varient considérablement en fonction de la charge.

Windows *Vérification du fonctionnement de MSCS*

Procédez comme suit pour vous assurer que vous disposez d'un cluster MSCS en cours d'exécution.

Les descriptions de tâche commençant par «[Création d'un gestionnaire de files d'attente à utiliser avec MSCS](#)», à la [page 532](#) supposent que vous disposez d'un cluster MSCS en cours d'exécution dans lequel

vous pouvez créer, migrer et détruire des ressources. Si vous souhaitez vous assurer que vous disposez d'un tel cluster:

1. A l'aide de l'administrateur de cluster MSCS, créez un groupe.
2. Dans ce groupe, créez une instance d'une ressource d'application générique en spécifiant l'horloge système (nom de chemin C:\winnt\system32\clock.exe et répertoire de travail de C:\).
3. Assurez-vous que vous pouvez mettre la ressource en ligne, que vous pouvez déplacer le groupe qui la contient vers l'autre noeud et que vous pouvez mettre la ressource hors ligne.

Windows Démarrage manuel et MSCS

Pour un gestionnaire de files d'attente géré par MSCS, vous devez définir l'attribut de démarrage sur manuel. Ainsi, le support IBM MQ MSCS peut redémarrer le service MQSeries sans démarrer immédiatement le gestionnaire de files d'attente.

La prise en charge de IBM MQ MSCS doit pouvoir redémarrer le service pour qu'il puisse effectuer la surveillance et le contrôle, mais doit rester en contrôle sur les gestionnaires de files d'attente en cours d'exécution et sur les machines. Pour plus d'informations, voir [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.

Windows MSCS et gestionnaires de files d'attente

Remarques concernant les gestionnaires de files d'attente lors de l'utilisation de MSCS.

Création d'un gestionnaire de files d'attente correspondant sur l'autre noeud

Pour que la mise en cluster fonctionne avec IBM MQ, vous avez besoin d'un gestionnaire de files d'attente identique sur le noeud B pour chacun d'eux sur le noeud A. Toutefois, vous n'avez pas besoin de créer explicitement la seconde. Vous pouvez créer ou préparer un gestionnaire de files d'attente sur un noeud, le déplacer sur l'autre noeud, comme décrit dans [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533, et il est entièrement dupliqué sur ce noeud.

Gestionnaires de files d'attente par défaut

N'utilisez pas de gestionnaire de files d'attente par défaut sous le contrôle de MSCS. Un gestionnaire de files d'attente ne possède pas de propriété qui en fait la valeur par défaut ; IBM MQ conserve son propre enregistrement distinct. Si vous déplacez un gestionnaire de files d'attente défini comme valeur par défaut sur l'autre ordinateur lors de la reprise en ligne, il ne devient pas la valeur par défaut. Faites en sorte que toutes vos applications fassent référence à des gestionnaires de files d'attente spécifiques par leur nom.

Suppression d'un gestionnaire de files d'attente

Une fois qu'un gestionnaire de files d'attente a déplacé le noeud, ses détails existent dans le registre sur les deux ordinateurs. Lorsque vous souhaitez le supprimer, faites-le normalement sur un ordinateur, puis exécutez l'utilitaire décrit dans [«Prise en charge des programmes utilitaires MSCS»](#), à la page 546 pour nettoyer le registre sur l'autre ordinateur.

Prise en charge des gestionnaires de files d'attente existants

Vous pouvez placer un gestionnaire de files d'attente existant sous le contrôle de MSCS, à condition que vous puissiez placer les fichiers journaux et les fichiers de file d'attente de votre gestionnaire de files d'attente sur un disque qui se trouve sur le bus SCSI partagé entre les deux machines (voir [Figure 71](#), à la page 530). Vous devez brièvement mettre le gestionnaire de files d'attente hors ligne lors de la création de la ressource MSCS.

Si vous souhaitez créer un gestionnaire de files d'attente, créez-le indépendamment de MSCS, testez ce dernier, puis placez-le sous le contrôle de MSCS. Voir :

- [«Création d'un gestionnaire de files d'attente à utiliser avec MSCS»](#), à la page 532

- [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS», à la page 533](#)
- [«Placement d'un gestionnaire de files d'attente sous contrôle MSCS», à la page 535](#)

Indiquer à MSCS les gestionnaires de files d'attente à gérer

Vous choisissez les gestionnaires de files d'attente placés sous le contrôle de MSCS à l'aide de l'administrateur de cluster MSCS pour créer une instance de ressource pour chaque gestionnaire de files d'attente de ce type. Ce processus vous présente une liste de ressources à partir de laquelle vous pouvez sélectionner le gestionnaire de files d'attente que vous souhaitez que cette instance gère.

Fichiers journaux du gestionnaire de files d'attente

Lorsque vous déplacez un gestionnaire de files d'attente vers un stockage MSCS, vous déplacez ses fichiers journaux et de données vers un disque partagé (pour un exemple, voir [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS», à la page 533](#)).

Il est conseillé avant de déplacer, d'arrêter le gestionnaire de files d'attente proprement et d'effectuer une sauvegarde intégrale des fichiers de données et des fichiers journaux.

Utilisation de plusieurs gestionnaires de files d'attente

La prise en charge de IBM MQ MSCS permet d'exécuter plusieurs gestionnaires de files d'attente sur chaque machine et de placer des gestionnaires de files d'attente individuels sous le contrôle de MSCS.

Windows *Toujours utiliser MSCS pour gérer les clusters*

N'essayez pas d'effectuer des opérations de démarrage et d'arrêt directement sur un gestionnaire de files d'attente sous le contrôle de MSCS, à l'aide des commandes de contrôle ou de IBM MQ Explorer. A la place, utilisez MSCS Cluster Administrator pour mettre le gestionnaire de files d'attente en ligne ou le mettre hors ligne.

L'utilisation de l'administrateur de cluster MSCS permet en partie d'éviter toute confusion due au fait que MSCS signale que le gestionnaire de files d'attente est hors ligne, alors qu'en fait vous l'avez démarré en dehors du contrôle de MSCS. Plus sérieusement, l'arrêt d'un gestionnaire de files d'attente sans utiliser MSCS est détecté par MSCS en tant qu'incident, ce qui déclenche la reprise en ligne sur l'autre noeud.

Windows *Travail en mode actif / actif dans MSCS*

Les deux ordinateurs du cluster MSCS peuvent exécuter des gestionnaires de files d'attente en mode Actif / Actif. Vous n'avez pas besoin d'avoir une machine complètement inactive agissant comme une machine de secours (mais vous pouvez, si vous le souhaitez, en mode actif / passif).

Si vous prévoyez d'utiliser les deux machines pour exécuter la charge de travail, fournissez à chacune d'elles une capacité suffisante (processeur, mémoire, stockage secondaire) pour exécuter la totalité de la charge de travail du cluster à un niveau de performances satisfaisant.

Remarque : Si vous utilisez MSCS avec Microsoft Transaction Server (COM +), vous **ne pouvez pas** utiliser le mode Actif / Actif. En effet, pour utiliser IBM MQ avec MSCS et COM +:

- Les composants d'application qui utilisent la prise en charge de IBM MQ COM + doivent s'exécuter sur le même ordinateur que le composant DTC (Distributed Transaction Coordinator), qui fait partie de COM +.
- Le gestionnaire de files d'attente doit également s'exécuter sur le même ordinateur.
- Le code défaut doit être configuré en tant que ressource MSCS et ne peut donc être exécuté que sur un seul des ordinateurs du cluster à la fois.

Windows *Commande PostOnline et commande PreOfflinedans MSCS*

Utilisez ces commandes pour intégrer la prise en charge de IBM MQ MSCS à d'autres systèmes. Vous pouvez les utiliser pour émettre des commandes IBM MQ , avec certaines restrictions.

Spécifiez ces commandes dans les paramètres pour une ressource de type IBM MQ MSCS. Vous pouvez les utiliser pour intégrer la prise en charge de IBM MQ MSCS à d'autres systèmes ou procédures. Par

exemple, vous pouvez spécifier le nom d'un programme qui envoie un message électronique, active un messenger de poche ou génère une autre forme d'alerte à capturer par un autre système de surveillance.

La commande PostOnline est appelée lorsque la ressource passe de l'état hors ligne à l'état en ligne ; la commande PreOffline est appelée pour passer de l'état en ligne à l'état hors ligne. Lorsqu'elles sont appelées, ces commandes sont exécutées par défaut à partir du répertoire système Windows . Etant donné que IBM MQ utilise un processus de moniteur de ressources 32 bits, sur les systèmes Windows 64 bits, il s'agit du répertoire \Windows\SysWOW64 et non du répertoire \Windows\system32 . Pour plus d'informations, voir la documentation Microsoft sur la redirection de fichier dans un environnement Windows x64 . Les deux commandes s'exécutent sous le compte utilisateur utilisé pour exécuter le service de cluster MSCS et sont appelées de manière asynchrone. Le support IBM MQ MSCS n'attend pas qu'elles se terminent avant de continuer. Cela élimine tout risque qu'ils bloquent ou retardent d'autres opérations de cluster.

Vous pouvez également utiliser ces commandes pour émettre des commandes IBM MQ , par exemple pour redémarrer les canaux demandeurs. Toutefois, les commandes sont exécutées au moment où l'état du gestionnaire de files d'attente change, de sorte qu'elles ne sont pas destinées à exécuter des fonctions de longue durée et ne doivent pas émettre d'hypothèses sur l'état en cours du gestionnaire de files d'attente ; il est tout à fait possible que, immédiatement après la mise en ligne du gestionnaire de files d'attente, un administrateur ait émis une commande hors ligne.

Si vous souhaitez exécuter des programmes qui dépendent de l'état du gestionnaire de files d'attente, envisagez de créer des instances du type de ressource MSCS Generic Application , de les placer dans le même groupe MSCS que la ressource du gestionnaire de files d'attente et de les rendre dépendantes de la ressource du gestionnaire de files d'attente.

Windows *Utilisation de noeuds favoris dans MSCS*

Il peut être utile lorsque vous utilisez le mode actif / actif dans MSCS pour configurer un *noeud préférentiel* pour chaque gestionnaire de files d'attente. Cependant, en général, il est préférable de ne pas définir de noeud préférentiel mais de s'appuyer sur une reprise par restauration manuelle.

Contrairement à d'autres ressources relativement sans état, un gestionnaire de files d'attente peut prendre un certain temps pour basculer (ou revenir) d'un noeud à l'autre. Pour éviter des indisponibilités inutiles, testez le noeud récupéré avant d'y renvoyer un gestionnaire de files d'attente défaillant. Cela empêche l'utilisation du paramètre de reprise par restauration *immediate* . Vous pouvez configurer la reprise par restauration entre certaines heures de la journée.

La route la plus sûre consiste probablement à replacer manuellement le gestionnaire de files d'attente sur le noeud requis, lorsque vous êtes certain que le noeud est entièrement récupéré. Cela exclut l'utilisation de l'option *preferred node* .

Windows *Erreurs COM + lors de l'installation sur MSCS*

Lorsque vous installez IBM MQ sur un cluster MSCS nouvellement installé, il se peut qu'une erreur liée à la source COM + et à l'ID événement 4691 soit signalée dans le journal des événements d'application.

Cela signifie que vous essayez d'exécuter IBM MQ dans un environnement Microsoft Cluster Server (MSCS) alors que le composant Microsoft Distributed Transaction Coordinator (MSDTC) n'a pas été configuré pour s'exécuter dans un environnement de ce type. Pour plus d'informations sur la configuration de MSDTC dans un environnement de cluster, voir la documentation Microsoft .

Windows *Prise en charge des programmes utilitaires MSCS*

Liste de la prise en charge par IBM MQ des programmes utilitaires Microsoft Cluster Service (MSCS) que vous pouvez exécuter à l'invite de commande.

Remarque : Depuis la Windows Server 2016, le nouveau nom de Microsoft Cluster Service (MSCS) est Windows Server Failover Clustering (WSFC).

La prise en charge de MSCS par IBM MQ inclut les programmes utilitaires suivants:

Enregistrer / désenregistrer le type de ressource
haregtyp.exe

Après avoir *désenregistré* le type de ressource MSCS IBM MQ , vous ne pouvez plus créer de ressources de ce type. MSCS ne vous permet pas d'annuler l'enregistrement d'un type de ressource si vous avez toujours des instances de ce type dans le cluster:

1. A l'aide de l'administrateur de cluster MSCS, arrêtez tous les gestionnaires de files d'attente qui s'exécutent sous le contrôle de MSCS, en les mettant hors ligne comme décrit dans [«Mise hors ligne d'un gestionnaire de files d'attente à partir de MSCS»](#), à la page 542.
2. A l'aide de l'administrateur de cluster MSCS, supprimez les instances de ressource.
3. A l'invite de commande, désenregistrez le type de ressource en entrant la commande suivante:

```
haredtyp /u
```

Si vous souhaitez *enregistrer* le type (ou le réenregistrer ultérieurement), entrez la commande suivante à l'invite de commande:

```
haredtyp /r
```

Après avoir correctement enregistré les bibliothèques MSCS, vous devez réamorcer le système si vous ne l'avez pas fait depuis l'installation de IBM MQ.

Déplacement d'un gestionnaire de files d'attente vers le stockage MSCS

hamvmqm.exe

Voir [«Transfert d'un gestionnaire de files d'attente vers le stockage MSCS»](#), à la page 533.

Suppression d'un gestionnaire de files d'attente d'un noeud

hadl1mqm.exe

Prenons le cas où vous avez un gestionnaire de files d'attente dans votre cluster, qu'il a été déplacé d'un noeud à un autre et que vous souhaitez le détruire. Utilisez l'explorateur IBM MQ pour le supprimer sur le noeud où il se trouve actuellement. Les entrées de registre associées existent toujours sur l'autre ordinateur. Pour les supprimer, entrez la commande suivante à l'invite de cet ordinateur:

```
hadl1mqm /m qmname
```

où qmname est le nom du gestionnaire de files d'attente à supprimer.

Vérifier et sauvegarder les détails de configuration

amqmsysn.exe

Cet utilitaire présente une boîte de dialogue présentant les détails complets de votre configuration de support IBM MQ MSCS, comme cela peut être demandé si vous appelez le support IBM . Il existe une option permettant de sauvegarder les détails dans un fichier.

Multi

Gestionnaires de files d'attente multi-instance

Les gestionnaires de files d'attente multi-instance sont des instances du même gestionnaire de files d'attente configurées sur des serveurs différents. Une instance du gestionnaire de files d'attente est définie en tant qu'instance active et une autre instance est définie en tant qu'instance de secours. En cas de défaillance de l'instance active, le gestionnaire de files d'attente multi-instance redémarre automatiquement sur le serveur de secours.

Exemple de configuration de gestionnaire de files d'attente multi-instance

La [Figure 72](#), à la page 548 présente un exemple de configuration multi-instance pour le gestionnaire de files d'attente QM1. IBM MQ est installé sur deux serveurs, dont l'un est une unité de secours. Un gestionnaire de files d'attente, QM1, a été créé. Une instance de QM1 est active et s'exécute sur un serveur. L'autre instance de QM1 s'exécute en veille sur l'autre serveur, sans traitement actif, mais prête à prendre le relais de l'instance active de QM1, si l'instance active échoue. (Il ne peut y avoir qu'une seule

instance active et une seule instance de secours du gestionnaire de files d'attente dans une configuration multi-instance.)

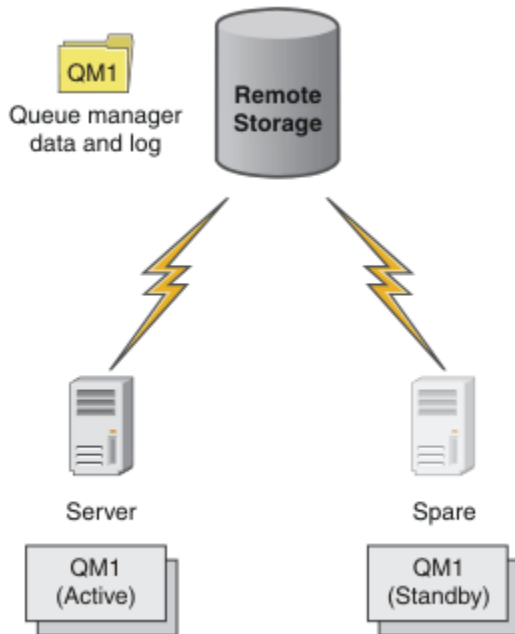


Figure 72. Gestionnaire de files d'attente multi-instance

Lorsque vous prévoyez d'utiliser un gestionnaire de files d'attente en tant que gestionnaire de files d'attente multi-instance, créez un gestionnaire de files d'attente unique sur l'un des serveurs à l'aide de la commande **crtmqm**, en plaçant ses données de gestionnaire de files d'attente et ses journaux dans le stockage réseau partagé. Sur l'autre serveur, au lieu de créer à nouveau le gestionnaire de files d'attente, utilisez la commande **addmqinf** pour créer une référence aux données du gestionnaire de files d'attente et aux journaux sur le stockage réseau.

Vous pouvez maintenant exécuter le gestionnaire de files d'attente à partir de l'un des serveurs. Chacun des serveurs fait référence aux mêmes données et journaux de gestionnaire de files d'attente ; il n'y a qu'un seul gestionnaire de files d'attente et il est actif sur un seul serveur à la fois.

Le gestionnaire de files d'attente peut s'exécuter en tant que gestionnaire de files d'attente à instance unique ou en tant que gestionnaire de files d'attente à instances multiples. Dans les deux cas, une seule instance du gestionnaire de files d'attente est en cours d'exécution et traite les demandes. La différence est que lors de l'exécution en tant que gestionnaire de files d'attente multi-instance, le serveur qui n'exécute pas l'instance active du gestionnaire de files d'attente s'exécute en tant qu'instance de secours, prête à prendre le relais de l'instance active automatiquement en cas de défaillance du serveur actif.

Le seul contrôle que vous avez sur l'instance qui devient active en premier est l'ordre dans lequel vous démarrez le gestionnaire de files d'attente sur les deux serveurs. La première instance à acquérir des verrous en lecture / écriture sur les données du gestionnaire de files d'attente devient l'instance active.

Vous pouvez permuter l'instance active sur l'autre serveur, une fois qu'elle a démarré, en arrêtant l'instance active à l'aide de l'option de commutation pour transférer le contrôle sur la base de données de secours.

L'instance active de QM1 dispose d'un accès exclusif aux données du gestionnaire de files d'attente partagées et aux dossiers des journaux lorsqu'elle est en cours d'exécution. L'instance de secours de QM1 détecte l'échec de l'instance active et devient l'instance active. Il reprend les données et les journaux QM1 dans l'état qu'ils ont été laissés par l'instance active et accepte les reconnections des clients et des canaux.

L'instance active peut échouer pour diverses raisons qui entraînent la prise en charge de la base de données de secours:

- Echec du serveur hébergeant l'instance de gestionnaire de files d'attente active.
- Echec de la connectivité entre le serveur hébergeant l'instance de gestionnaire de files d'attente active et le système de fichiers.
- Absence de réponse des processus du gestionnaire de files d'attente, détectée par IBM MQ, qui arrête ensuite le gestionnaire de files d'attente.

Vous pouvez ajouter les informations de configuration du gestionnaire de files d'attente à plusieurs serveurs et choisir deux serveurs à exécuter en tant que paire active / de secours. Il existe une limite de deux instances au total. Vous ne pouvez pas avoir deux instances de secours et une instance active.

Composants supplémentaires nécessaires à la génération d'une solution à haute disponibilité

Un gestionnaire de files d'attente multi-instance fait partie d'une solution à haute disponibilité. Vous avez besoin de composants supplémentaires pour créer une solution à haute disponibilité utile.

- Reconnexion du client et du canal pour transférer des connexions IBM MQ à l'ordinateur qui prend le relais de l'exécution de l'instance de gestionnaire de files d'attente active.
- Système de fichiers réseau partagé hautes performances (NFS) qui gère correctement les verrous et fournit une protection contre les pannes de support et de serveur de fichiers.

Important : Vous devez arrêter toutes les instances de gestionnaire de files d'attente multi-instance qui s'exécutent dans votre environnement avant de pouvoir effectuer une maintenance sur l'unité NFS . Vérifiez que vous disposez de sauvegardes de configuration de gestionnaire de files d'attente à récupérer en cas d'échec NFS .

- Des réseaux résilients et des alimentations électriques pour éliminer les points de défaillance uniques de l'infrastructure de base.
- Applications qui tolèrent la reprise en ligne. En particulier, vous devez prêter une attention particulière au comportement des applications transactionnelles et aux applications qui parcourent les files d'attente IBM MQ .
- Surveillance et gestion des instances actives et de secours pour s'assurer qu'elles sont en cours d'exécution et pour redémarrer les instances actives qui ont échoué. Bien que les gestionnaires de files d'attente multi-instance redémarrent automatiquement, vous devez vous assurer que vos instances de secours sont en cours d'exécution, prêtes à prendre le relais et que les instances ayant échoué sont reconnectées en tant que nouvelles instances de secours.

IBM MQ MQI clients et les canaux se reconnectent automatiquement au gestionnaire de files d'attente de secours lorsqu'il devient actif. Vous trouverez plus d'informations sur la reconnexion et sur les autres composants d'une solution à haute disponibilité dans les rubriques connexes. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java.

Plateformes prises en charge

Vous pouvez créer un gestionnaire de files d'attente multi-instance sur n'importe quel système Multiplatforms.

La reconnexion automatique du client est prise en charge pour les clients MQI.

Création d'un gestionnaire de files d'attente multi-instance

Créez un gestionnaire de files d'attente multi-instance, créez le gestionnaire de files d'attente sur un serveur et configurez IBM MQ sur un autre serveur. Les gestionnaires de files d'attente multi-instance partagent les données et les journaux du gestionnaire de files d'attente.

La plupart des efforts nécessaires à la création d'un gestionnaire de files d'attente multi-instance consiste à configurer les données et les fichiers journaux du gestionnaire de files d'attente partagé. Vous devez créer des répertoires partagés sur le stockage réseau et les mettre à la disposition des autres serveurs à l'aide de partages réseau. Ces tâches doivent être effectuées par une personne disposant des droits d'administration, telle que *root* sur les systèmes AIX and Linux . Les étapes à suivre sont les suivantes :

1. Créez les partages pour les données et les fichiers journaux.
2. Créez le gestionnaire de files d'attente sur un serveur.
3. Exécutez la commande **dspmqlinf** sur le premier serveur pour collecter les données de configuration du gestionnaire de files d'attente et les copier dans le presse-papiers.
4. Exécutez la commande **addmqinf** avec les données copiées pour créer la configuration du gestionnaire de files d'attente sur le second serveur.

Vous n'exécutez pas **crtmqm** pour recréer le gestionnaire de files d'attente sur le second serveur.

Contrôle d'accès aux fichiers

Vous devez veiller à ce que l'utilisateur et le groupe mqm sur tous les autres serveurs soient autorisés à accéder aux partages.

Sous AIX and Linux, vous devez rendre les `uid` et les `gid` de mqm identiques sur tous les systèmes. Vous devrez peut-être éditer `/etc/passwd` sur chaque système pour définir un `uid` et un `gid` communs pour mqm, puis réamorcer votre système.

Sous Microsoft Windows, l'ID utilisateur qui exécute les processus du gestionnaire de files d'attente doit disposer des droits de contrôle complets sur les répertoires contenant les données du gestionnaire de files d'attente et les fichiers journaux. Vous pouvez configurer l'autorisation de deux manières:

1. Créez un gestionnaire de files d'attente avec un groupe global comme autre principal de sécurité. Autorisez le groupe global à avoir un accès de contrôle complet aux répertoires contenant les données du gestionnaire de files d'attente et les fichiers journaux ; voir [«Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows»](#), à la page 579. Faites de l'ID utilisateur qui exécute le gestionnaire de files d'attente un membre du groupe global. Vous ne pouvez pas définir un utilisateur local comme membre d'un groupe global. Par conséquent, les processus du gestionnaire de files d'attente doivent s'exécuter sous un ID utilisateur de domaine. L'ID utilisateur de domaine doit être membre du groupe local mqm. La tâche [«Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows»](#), à la page 553 explique comment configurer un gestionnaire de files d'attente multi-instance à l'aide de fichiers sécurisés de cette manière.
2. Créez un gestionnaire de files d'attente sur le contrôleur de domaine, de sorte que le groupe mqm local ait une portée de domaine, "domain local". Sécurisez le partage de fichiers avec le domaine local mqmet exécutez les processus de gestionnaire de files d'attente sur toutes les instances d'un gestionnaire de files d'attente sous le même groupe mqm local de domaine. La tâche [«Création d'un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine Windows»](#), à la page 569 explique comment configurer un gestionnaire de files d'attente multi-instance à l'aide de fichiers sécurisés de cette manière.

Informations de configuration

Configurez autant d'instances de gestionnaire de files d'attente que nécessaire en modifiant les informations de configuration de gestionnaire de files d'attente IBM MQ sur chaque serveur. Chaque serveur doit avoir la même version de IBM MQ installée à un niveau de correctif compatible. Les commandes **dspmqlinf** et **addmqinf** vous aident à configurer les instances de gestionnaire de files d'attente supplémentaires. Vous pouvez également éditer directement les fichiers `mqs.ini` et `qm.ini`. Les rubriques [«Création d'un gestionnaire de files d'attente multi-instance sous Linux»](#), à la page 593, [«Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows»](#), à la page 553 et [«Création d'un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine Windows»](#), à la page 569 sont des exemples illustrant la configuration d'un gestionnaire de files d'attente multi-instance.

Sur les systèmes AIX, Linux, and Windows, vous pouvez partager un seul fichier `mqs.ini` en le plaçant sur le partage de réseau et en définissant la variable d'environnement **AMQ_MQS_INI_LOCATION** pour qu'elle pointe vers celui-ci.

Restrictions

1. Configurez plusieurs instances du même gestionnaire de files d'attente uniquement sur des serveurs ayant le même système d'exploitation, la même architecture et le même ordre d'octets. Par exemple, les deux machines doivent être 32 bits ou 64 bits.
2. Toutes les installations IBM MQ doivent être au niveau d'édition 7.0.1 ou supérieur.
3. En général, les installations actives et de secours sont gérées au même niveau de maintenance. Consultez les instructions de maintenance pour chaque mise à niveau afin de vérifier si vous devez mettre à niveau toutes les installations ensemble.

Notez que les niveaux de maintenance des gestionnaires de files d'attente actifs et passifs doivent être identiques.

4. Partagez les données et les journaux du gestionnaire de files d'attente uniquement entre les gestionnaires de files d'attente configurés avec le même utilisateur, le même groupe et le même mécanisme de contrôle d'accès IBM MQ . **IBM i** Par exemple, le partage de réseau configuré sur un serveur Linux peut contenir des données de gestionnaire de files d'attente et des journaux distincts pour les gestionnaires de files d'attente AIX and Linux , mais ne peut pas contenir les données de gestionnaire de files d'attente utilisées par IBM i.

IBM i Vous pouvez créer plusieurs partages sur le même stockage en réseau pour IBM i et pour les systèmes AIX and Linux tant que les partages sont différents. Vous pouvez donner différents partages différents propriétaires. La restriction est une conséquence des différents noms utilisés pour les utilisateurs et les groupes IBM MQ entre AIX and Linux et IBM i. Le fait que l'utilisateur et le groupe puissent avoir les mêmes uid et gid n'assouplit pas la restriction.

5. Sur les systèmes AIX and Linux , configurez le système de fichiers partagé sur le stockage en réseau avec un montage matériel, interruptible, plutôt qu'un montage logiciel . Un montage interruptible à froid force le gestionnaire de files d'attente à se bloquer jusqu'à ce qu'il soit interrompu par un appel système. Les montages logiciels ne garantissent pas la cohérence des données après une défaillance du serveur.
6. Les répertoires de journaux et de données partagés ne peuvent pas être stockés sur un système de fichiers FAT ou NFSv3 . Pour les gestionnaires de files d'attente multi-instance sous Windows, le stockage en réseau doit être accessible par le protocole CIFS (Common Internet File System) utilisé par les réseaux Windows .
7. **z/OS** z/OS ne prend pas en charge les gestionnaires de files d'attente multi-instance. Utilisez des groupes de partage de files d'attente.

Les clients reconnectables fonctionnent avec les gestionnaires de files d'attente z/OS .

Windows Domaines et gestionnaires de files d'attente multi-instances Windows

Un gestionnaire de files d'attente multi-instance sous Windows requiert le partage de ses données et de ses journaux. Le partage doit être accessible à toutes les instances du gestionnaire de files d'attente s'exécutant sur des serveurs ou des postes de travail différents. Configurez les gestionnaires de files d'attente et partagez-les dans le cadre d'un domaine Windows . Le gestionnaire de files d'attente peut s'exécuter sur un poste de travail ou un serveur de domaine ou sur le contrôleur de domaine.

Important : Par défaut, le fonctionnement des ordinateurs avec Windows 10 version 1607 et Windows Server 2016 devient plus restrictifs qu'avec les versions antérieures de Windows.

Ce changement restreint les clients autorisés à effectuer des appels distants au gestionnaire de comptes de sécurité (SAM) et peut avoir un impact sur IBM MQ, notamment en empêchant les gestionnaires de files d'attente de démarrer. L'accès à SAM est essentiel pour le fonctionnement de IBM MQ lorsque IBM MQ est configuré en tant que compte de domaine.

Avant de configurer un gestionnaire de files d'attente multi-instance, lisez «Fichiers et répertoires de données et de journaux du gestionnaire de files d'attente non partagés sécurisés sous Windows», à la page 583 et «Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows», à la page 579 pour savoir comment contrôler l'accès aux données

et aux fichiers journaux du gestionnaire de files d'attente. Les rubriques sont éducatives ; si vous souhaitez accéder directement à la configuration de répertoires partagés pour un gestionnaire de files d'attente multi-instance dans un domaine Windows , voir [«Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows»](#), à la page 553.

Exécuter un gestionnaire de files d'attente multi-instance sur des postes de travail ou des serveurs de domaine

Depuis la IBM WebSphere MQ 7.1, les gestionnaires de files d'attente multi-instance s'exécutent sur un poste de travail ou un serveur membre d'un domaine. Pour exécuter un gestionnaire de files d'attente multi-instance sous Windows, vous avez besoin d'un contrôleur de domaine, d'un serveur de fichiers et de deux postes de travail ou serveurs exécutant le même gestionnaire de files d'attente connectés au même domaine.

La modification qui permet d'exécuter un gestionnaire de files d'attente multi-instance sur n'importe quel serveur ou poste de travail d'un domaine est que vous pouvez désormais créer un gestionnaire de files d'attente avec un groupe de sécurité supplémentaire. Le groupe de sécurité supplémentaire est transmis dans la commande `crtmqm` , dans le paramètre `-a` . Vous sécurisez les répertoires qui contiennent les données et les journaux du gestionnaire de files d'attente avec le groupe. L'ID utilisateur qui exécute les processus du gestionnaire de files d'attente doit être membre de ce groupe. Lorsque le gestionnaire de files d'attente accède aux répertoires, Windows vérifie les droits dont dispose l'ID utilisateur pour accéder aux répertoires. En attribuant à la fois la portée de domaine du groupe et de l'ID utilisateur, l'ID utilisateur exécutant les processus du gestionnaire de files d'attente possède des données d'identification provenant du groupe global. Lorsque le gestionnaire de files d'attente s'exécute sur un serveur différent, l'ID utilisateur exécutant les processus du gestionnaire de files d'attente peut avoir les mêmes données d'identification. Il n'est pas nécessaire que l'ID utilisateur soit identique. Il doit être membre du groupe de sécurité alternatif, ainsi que membre du groupe `mqm` local.

Pour plus de détails sur la création d'un gestionnaire de files d'attente multi-instance, voir [«Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows»](#), à la page 553 .

Plusieurs étapes sont requises pour configurer le domaine, ainsi que les serveurs de domaine et les postes de travail. Vous devez comprendre comment Windows autorise un gestionnaire de files d'attente à accéder à ses répertoires de données et de journaux. Si vous ne savez pas comment les processus de gestionnaire de files d'attente sont autorisés à accéder à leurs fichiers journaux et de données, lisez la rubrique [«Fichiers et répertoires de données et de journaux du gestionnaire de files d'attente non partagés sécurisés sous Windows»](#), à la page 583. La rubrique inclut deux tâches pour vous aider à comprendre les étapes requises. Les tâches sont [«Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm»](#), à la page 585 et [«Lecture et écriture de données et de fichiers journaux autorisés par un autre groupe de sécurité local»](#), à la page 589. Une autre rubrique, [«Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows»](#), à la page 579, explique comment sécuriser les répertoires partagés contenant les données du gestionnaire de files d'attente et les fichiers journaux avec le groupe de sécurité alternatif. La rubrique comprend quatre tâches: la configuration d'un domaine Windows , la création d'un partage de fichiers, l'installation de IBM MQ for Windows et la configuration d'un gestionnaire de files d'attente pour utiliser le partage. Les tâches sont les suivantes:

1. [«Création d'un domaine Active Directory et DNS sur Windows»](#), à la page 556.
2. [«Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows»](#), à la page 560.
3. [«Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows»](#), à la page 563.
4. [«Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale»](#), à la page 566.

Vous pouvez ensuite effectuer la tâche [«Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows»](#), à la page 553 à l'aide du domaine.

Effectuez ces tâches pour explorer la configuration d'un gestionnaire de files d'attente multi-instance avant de transférer vos connaissances vers un domaine de production.

Exécuter un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine

Les données du gestionnaire de files d'attente peuvent être sécurisées avec le groupe de domaine `mqm`. Comme l'explique la rubrique «Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows», à la page 579, vous ne pouvez pas partager des répertoires sécurisés avec le groupe `mqm` local sur des postes de travail ou des serveurs. Toutefois, sur les contrôleurs de domaine, tous les groupes et principaux ont une portée de domaine. Si vous installez IBM MQ for Windows sur un contrôleur de domaine, les données du gestionnaire de files d'attente et les fichiers journaux sont sécurisés avec le groupe `mqm` de domaine, qui peut être partagé. Suivez les étapes de la tâche «Création d'un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine Windows», à la page 569 pour configurer un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine.

Information associée

Gestion des autorisations et du contrôle d'accès

Utilisation des noeuds de cluster Windows Server en tant que contrôleurs de domaine

Windows *Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows*

Un exemple indique comment configurer un gestionnaire de files d'attente multi-instance sous Windows ou sur un poste de travail ou un serveur faisant partie d'un domaine Windows. Le serveur ne doit pas nécessairement être un contrôleur de domaine. La configuration illustre les concepts concernés plutôt que de traiter le niveau de production. L'exemple est basé sur Windows Server 2008. Les étapes peuvent varier d'une version à l'autre de Windows Server.

Dans une configuration de niveau production, il peut s'avérer nécessaire de l'ajuster en fonction d'un domaine existant. Par exemple, vous pouvez être amené à définir des groupes de domaines différents afin d'autoriser différents partages et de grouper les ID utilisateur qui exécutent les gestionnaires de files d'attente.

La configuration exemple est constituée de trois serveurs :

sun

Un contrôleur de domaine Windows Server 2008. Il possède le domaine *wmq.example.com* qui contient *Sun*, *marset* *venus*. A des fins de démonstration, il est également utilisé comme serveur de fichiers.

mars

Un serveur Windows Server 2008 utilisé en tant que premier serveur IBM MQ. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

venus

Un serveur Windows Server 2008 utilisé en tant que deuxième serveur IBM MQ. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

Sous Windows, vous n'avez pas besoin de vérifier le système de fichiers sur lequel vous prévoyez de stocker les données et les fichiers journaux du gestionnaire de files d'attente. La procédure de vérification Vérification du comportement du système de fichiers partagé, s'applique à AIX and Linux. Sous Windows, les vérifications aboutissent toujours.

Effectuez les étapes des tâches suivantes. Les tâches créent le contrôleur de domaine et le domaine, installent IBM MQ for Windows sur un serveur et créent le partage de fichiers pour les données et les

fichiers journaux. Si vous configurez un contrôleur de domaine existant, il peut s'avérer utile de tester les étapes sur un nouveau serveur Windows 2008. Vous pouvez adapter les étapes à votre domaine.

1. [«Création d'un domaine Active Directory et DNS sur Windows»](#), à la page 556.
2. [«Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows»](#), à la page 560.
3. [«Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows»](#), à la page 563.
4. [«Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale»](#), à la page 566.

Pourquoi et quand exécuter cette tâche

Cette tâche fait partie d'une séquence de tâches visant à configurer un contrôleur de domaine et deux serveurs dans le domaine de sorte qu'ils exécutent des instances d'un gestionnaire de files d'attente. Dans cette tâche, vous configurez un second serveur, *venus*, pour exécuter une autre instance du gestionnaire de files d'attente *QMGR*. Procédez comme suit pour créer la deuxième instance du gestionnaire de files d'attente, *QMGR*, et tester son fonctionnement.

Cette tâche est distincte des quatre tâches dans la section précédente. Elle contient les étapes permettant de convertir un gestionnaire de files d'attente mono-instance en gestionnaire de files d'attente multi-instance. Toutes les autres étapes sont communes aux gestionnaires de files d'attente mono ou multi-instance.

Procédure

1. Configurez un deuxième serveur de sorte qu'il exécute IBM MQ for Windows.
 - a) Suivez la procédure décrite dans la tâche [«Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows»](#), à la page 560 pour créer un deuxième serveur de domaine. Dans cette séquence de tâches, le second serveur est appelé *venus*.

Conseil : Créez la deuxième installation en utilisant les mêmes valeurs par défaut d'installation pour IBM MQ sur chacun des deux serveurs. Si les valeurs par défaut diffèrent, vous devrez peut-être personnaliser les variables `Prefix` et `InstallationName` dans la section *QMGR QueueManager* du fichier de configuration IBM MQ `mqmqs.ini`. Les variables désignent les chemins pouvant varier pour chaque installation et gestionnaire de files d'attente sur chaque serveur. Si les chemins restent identiques sur chaque serveur, il est plus simple de configurer un gestionnaire de files d'attente multi-instance.

2. Créez une deuxième instance de *QMGR* sur *venus*.
 - a) Si *QMGR* sur *mars* n'existe pas, effectuez la tâche [«Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale»](#), à la page 566 pour le créer.
 - b) Vérifiez que les valeurs des paramètres `Prefix` et `InstallationName` sont correctes pour *venus*.

Sous *mars*, exécutez la commande **dspmqrinf** :

```
dspmqrinf QMGR
```

Réponse du système :

```
QueueManager:  
Name=QMGR  
Directory=QMGR  
Prefix=C:\ProgramData\IBM\MQ  
DataPath=\\sun\wmq\data\QMGR  
InstallationName=Installation1
```

- c) Copiez le formulaire lisible par la machine de la section **QueueManager** vers le presse-papiers.
Sous *mars* , exécutez à nouveau la commande **dspmqlnf** avec le paramètre `-o command` .

```
dspmqlnf -o command QMGR
```

Réponse du système :

```
addmqinf -s QueueManager -v Name=QMGR  
-v Directory=QMGR -v Prefix="C:\ProgramData\IBM\MQ"  
-v DataPath=\\sun\wmq\data\QMGR
```

- d) Sous *venus* , exécutez la commande **addmqinf** à partir du presse-papiers pour créer une instance du gestionnaire de files d'attente sous *venus* .

Si nécessaire, ajustez la commande en fonction des différences dans les paramètres `Prefix` ou `InstallationName` .

```
addmqinf -s QueueManager -v Name=QMGR  
-v Directory=QMGR -v Prefix="C:\ProgramData\IBM\MQ"  
-v DataPath=\\sun\wmq\data\QMGR
```

IBM MQ configuration information added.

3. Démarrez le gestionnaire de files d'attente *QMGR* sur *venus* , en autorisant les instances de secours.

- a) Vérifiez que *QMGR* on *mars* est arrêté.

Sous *mars* , exécutez la commande **dspmq** :

```
dspmq -m QMGR
```

La réponse système dépend du mode d'arrêt du gestionnaire de files d'attente ; par exemple :

```
C:\Users\Administrator>dspmq -m QMGR  
QMNAME(QMGR) STATUS(Ended immediately)
```

- b) Sous *venus* , exécutez la commande **strmqm** pour démarrer *QMGR* en autorisant les bases de données de secours:

```
strmqm -x QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

Résultats

Pour tester le basculement du gestionnaire de files d'attente multi-instance, procédez comme suit :

1. Sous *mars*, exécutez la commande **strmqm** pour démarrer *QMGR* en autorisant les bases de données de secours:

```
strmqm -x QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
A standby instance of queue manager 'QMGR' has been started.  
The active instance is running elsewhere.
```

2. Sous *venus*, exécutez la commande **endmqm** :

```
endmqm -r -s -i QMGR
```

Réponse du système sous *venus*:

```
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ended, permitting switchover to  
a standby instance.
```

Et sous *mars*:

```
dspmq  
QMNAME(QMGR) STATUS(Running as standby)  
C:\Users\wmquser2>dspmq  
QMNAME(QMGR) STATUS(Running as standby)  
C:\Users\wmquser2>dspmq  
QMNAME(QMGR) STATUS(Running)
```

Que faire ensuite

Pour vérifier un gestionnaire de files d'attente multi-instance par le biais de plusieurs exemples de programme, voir [«Vérification du gestionnaire de files d'attente multi-instance sous Windows»](#), à la page 577.

Windows Création d'un domaine Active Directory et DNS sur Windows

Cette tâche crée le domaine *wmq.example.com* sur un contrôleur de domaine Windows 2008 appelé *sun*. Il configure le groupe global Domain *mqm* dans le domaine, avec les droits appropriés, et avec un seul utilisateur.

Dans une configuration de niveau production, il peut s'avérer nécessaire de l'ajuster en fonction d'un domaine existant. Par exemple, vous pouvez être amené à définir des groupes de domaines différents afin d'autoriser différents partages et de grouper les ID utilisateur qui exécutent les gestionnaires de files d'attente.

La configuration exemple est constituée de trois serveurs :

sun

Un contrôleur de domaine Windows Server 2008. Il possède le domaine *wmq.example.com* qui contient *Sun*, *marset* *venus*. A des fins de démonstration, il est également utilisé comme serveur de fichiers.

mars

Un serveur Windows Server 2008 utilisé en tant que premier serveur IBM MQ. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

venus

Un serveur Windows Server 2008 utilisé en tant que deuxième serveur IBM MQ. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

1. Les étapes de la tâche sont cohérentes avec un serveur Windows Server 2008 qui est installé mais qui n'est pas configuré avec des rôles. Si vous configurez un contrôleur de domaine existant, il peut s'avérer utile de tester les étapes sur un nouveau serveur Windows 2008. Vous pouvez adapter les étapes à votre domaine.

Pourquoi et quand exécuter cette tâche

Dans cette tâche, vous allez créer un domaine Active Directory et DNS sur un contrôleur de domaine. Vous devez ensuite le configurer pour qu'il soit prêt à installer IBM MQ sur d'autres serveurs et postes de travail qui rejoignent le domaine. Suivez cette tâche si vous n'êtes pas familiarisé avec l'installation et la configuration d'Active Directory pour créer un domaine Windows. Vous devez créer un domaine Windows afin de créer une configuration de gestionnaire de files d'attente multi-instance. La tâche n'est pas conçue pour vous aider à configurer un domaine Windows. Pour déployer des gestionnaires de files d'attente multi-instance dans un environnement de production, vous devez consulter la documentation Windows.

Lors de cette tâche, vous effectuez les étapes suivantes :

1. Installer Active Directory.
2. Ajouter un domaine.
3. Ajouter un domaine au DNS.
4. Créer le groupe global Domain *mqm* et lui attribuer les droits appropriés.
5. Ajouter un utilisateur et en faire un membre du groupe global Domain *mqm*.

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix. Elles accompagnent la tâche [«Domaines et gestionnaires de files d'attente multi-instances Windows»](#), à la page 551.

Pour les besoins de la tâche, le nom d'hôte du contrôleur de domaine est *sun* et les deux serveurs IBM MQ sont appelés *mars* et *venus*. Le domaine est appelé *wmq.example.com*. Vous pouvez remplacer tous les noms en italique dans la tâche par les noms de votre choix.

Procédure

1. Connectez-vous au contrôleur de domaine, *sun*, en tant qu'administrateur local ou `Workgroup`.
Si le serveur est déjà configuré en tant que contrôleur de domaine, vous devez vous connecter en tant qu'administrateur de domaine.
2. Exécutez l'assistant de service de domaine Active Directory.
 - a) Cliquez sur **Démarrer > Exécuter ...** Entrez `dcpromo` et cliquez sur **OK**.
Si les fichiers binaires Active Directory ne sont pas déjà installés, Windows installe les fichiers automatiquement.
3. Dans la première fenêtre de l'assistant, désélectionnez la case **Use advanced mode installation**. Cliquez sur **Suivant > Suivant**, puis sur **Créer un domaine dans une nouvelle forêt > Suivant**.
4. Entrez *wmq.example.com* dans la zone **Nom de domaine complet du domaine racine de la forêt**. Cliquez sur **Suivant**.

5. Dans la fenêtre Définir le niveau fonctionnel de la forêt, sélectionnez **Windows Server 2003**, ou version ultérieure, dans la liste **Niveaux fonctionnels de la forêt** > **Suivant**.

Le niveau le plus ancien de Windows Server pris en charge par IBM MQ est Windows Server 2003.

6. Facultatif : Dans la fenêtre Définition du niveau fonctionnel de domaine, sélectionnez **Windows Server 2003**, ou version ultérieure, dans la liste **Niveaux fonctionnels de domaine** > **Suivant**.

Cette étape est requise uniquement si vous définissez le niveau fonctionnel de la forêt sur **Windows Server 2003**.

7. La fenêtre Additional Domain Controller Options s'ouvre, avec **DNS server** sélectionné en tant qu'option supplémentaire. Cliquez sur **Next** et **Yes** pour fermer la fenêtre d'avertissement.

Conseil : Si un serveur DNS est déjà installé, cette option ne s'affiche pas. Si vous souhaitez suivre cette tâche précisément, supprimez tous les rôles de ce contrôleur de domaine et recommencez.

8. Laissez les répertoires Database, Log Files et SYSVOL inchangés ; cliquez sur **Suivant**.
9. Entrez un mot de passe dans la zone **Password** et **Confirm password** dans la fenêtre Directory Services Restore Mode Administrator Password. Cliquez sur **Suivant** > **Suivant**. Sélectionnez **Reboot on completion** dans la fenêtre de l'assistant final.
10. Lors du réamorçage du contrôleur de domaine, connectez-vous en tant que *wmq\Administrator*.

Le gestionnaire de serveur démarre automatiquement.

11. Ouvrez le dossier *wmq.example.com\Users*

- a) Ouvrez **Server Manager** > **Rôles** > **Active Directory Services de domaine** > *wmq.example.com* > **Utilisateurs**.

12. Cliquez avec le bouton droit de la souris sur **Utilisateurs** > **Nouveau** > **Groupe**.

- a) Entrez un nom de groupe dans la zone **Group name**.

Remarque : Le nom du groupe préféré est `Domain_mqm`. Tapez-le exactement comme indiqué.

- L'appel du groupe `Domain_mqm` modifie le comportement de l'Prepare IBM MQ Wizard sur un poste de travail ou un serveur du domaine. Il permet automatiquement à l'Prepare IBM MQ Wizard d'ajouter le groupe `Domain_mqm` local du groupe `mqm` dans chaque nouvelle installation d'IBM MQ dans le domaine.
- Vous pouvez installer des postes de travail ou des serveurs dans un domaine sans groupe global `Domain_mqm`. Si vous le faites, vous devez définir un groupe ayant les mêmes propriétés que le groupe `Domain_mqm`. Vous devez faire de ce groupe ou des utilisateurs qui en sont membres un membre du groupe `mqm` local partout où IBM MQ est installé dans un domaine. Vous pouvez placer les utilisateurs de domaine dans plusieurs groupes. Créez plusieurs groupes de domaine, chaque groupe correspondant à un ensemble d'installations que vous souhaitez gérer séparément. Séparez les utilisateurs de domaine, selon les installations qu'ils gèrent, dans des groupes de domaine différents. Ajoutez chaque groupe de domaine au groupe `mqm` local des différentes installations IBM MQ. Seuls les utilisateurs de domaine des groupes de domaines qui sont membres d'un groupe `mqm` local spécifique peuvent créer, administrer et exécuter des gestionnaires de files d'attente pour cette installation.
- L'utilisateur de domaine que vous nommez lors de l'installation de IBM MQ sur un poste de travail ou un serveur dans un domaine doit être membre du groupe `Domain_mqm` ou d'un autre groupe que vous avez défini avec les mêmes propriétés que le groupe `Domain_mqm`.

- b) Conservez la valeur **Global** sélectionnée pour **Group scope** ou remplacez-la par **Universal**. Conservez la valeur **Security** sélectionnée pour **Group type**. Cliquez sur **OK**.

13. Ajoutez les droits **Allow Read group membership** et **Allow Read groupMembershipSAM** aux droits du groupe global `Domain_mqm`.

- a) Dans la barre d'actions du gestionnaire de serveurs, cliquez sur **View** > **Advanced features**

- b) Dans l'arborescence de navigation du gestionnaire de serveur, cliquez sur **Users**

- c) Dans la fenêtre Utilisateurs, cliquez avec le bouton droit de la souris sur **Domain_mqm** > **Propriétés**

- d) Cliquez sur **Sécurité > Avancé > Ajouter ...**. Entrez Domain mqm et cliquez sur **Check names > OK**.

La zone **Nom** est préremplie avec la chaîne Domain mqm (*domain name*Domain mqm).

- e) Cliquez sur **Properties**. Dans la liste **Apply to**, sélectionnez **Descendant User Objects**.
f) Dans la liste **Droits d'accès**, cochez les cases **Lire l'appartenance au groupe** et **Lire groupMembershipSAM Autoriser**; cliquez sur **OK > Appliquer > OK > OK**.

14. Ajoutez au moins deux utilisateurs au groupe global Domain mqm.

Un utilisateur, *wmqluser1* dans l'exemple, exécute le service IBM MQ et l'autre utilisateur, *wmqluser2*, est utilisé de manière interactive.

Un utilisateur de domaine est requis pour créer un gestionnaire de files d'attente qui utilise le groupe de sécurité alternatif dans une configuration de domaine. Cela ne suffit pas que l'ID utilisateur soit un administrateur, bien que l'administrateur dispose des droits appropriés pour exécuter la commande **crtmqm**. L'utilisateur de domaine, qui pourrait être un administrateur, doit être membre du groupe local mqm ainsi que du groupe de sécurité alternatif.

Dans cet exemple, vous définissez *wmqluser1* et *wmqluser2* en tant que membres du groupe global Domain mqm. Prepare IBM MQ Wizard configure automatiquement Domain mqm en tant que membre du groupe mqm local où l'assistant est exécuté.

Vous devez fournir un utilisateur différent pour exécuter le service IBM MQ pour chaque installation d'IBM MQ sur un seul ordinateur. Vous pouvez réutiliser les mêmes utilisateurs sur différents ordinateurs.

- a) Dans l'arborescence de navigation de Server Manager, cliquez sur **Utilisateurs > Nouveau > Utilisateur**
b) Dans la fenêtre Nouvel objet-Utilisateur, entrez *wmqluser1* dans la zone **Nom de connexion de l'utilisateur**. Entrez *WebSphere* dans la zone **Prénom** et *MQ1* dans la zone **Nom**. Cliquez sur **Suivant**.
c) Entrez un mot de passe dans les zones **Password** et **Confirm password**, puis désélectionnez l'option **User must change password at next logon**. Cliquez sur **Suivant > Terminer**.
d) Dans la fenêtre Utilisateurs, cliquez avec le bouton droit de la souris sur **WebSphere MQ > Ajouter à un groupe ...**. Entrez Domain mqm et cliquez sur **Vérifier les noms > OK > OK**.
e) Répétez les étapes a à d pour ajouter *WebSphere MQ2* en tant que *wmqluser2*.

15. Exécutez IBM MQ en tant que service.

Si vous avez besoin d'exécuter IBM MQ en tant que service et d'accorder ensuite à l'utilisateur de domaine (obtenu de l'administrateur de domaine) l'accès à l'exécution en tant que service, procédez comme suit :

- a) Cliquez sur **Démarrer > Exécuter...**
Tapez la commande `secpol.msc` et cliquez sur **OK**.
b) Ouvrez **Paramètres de sécurité > Stratégies locales > Affectation des droits utilisateur**.
Dans la liste des règles, cliquez avec le bouton droit de la souris sur **Ouvrir une session en tant que service > Propriétés**.
c) Cliquez sur **Ajouter un utilisateur ou un groupe...**
Entrez le nom de l'utilisateur que vous avez obtenu de l'administrateur de domaine et cliquez sur **Vérifier les noms**.
d) A l'invite éventuelle d'une fenêtre de sécurité Windows, entrez le nom d'utilisateur et le mot de passe d'un utilisateur de compte ou d'un administrateur ayant des droits d'accès suffisants et cliquez sur **OK > Appliquer > OK**.
Fermez la fenêtre Stratégie de sécurité locale.

Remarque : Sous Windows Server 2008 et Windows Server 2012, le contrôle de compte utilisateur (UAC) est activé par défaut.

Cette fonction restreint les actions pouvant être menées par les utilisateurs sur certaines fonctions du système d'exploitation, même s'ils sont membres du groupe administrateur. Vous devez prendre les mesures appropriées pour contourner cette restriction.

Que faire ensuite

Exécutez la tâche suivante, «[Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows](#)», à la page 560.

Tâches associées

Windows [Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows](#)

Windows [Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows](#)

Windows [Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale](#)

Windows *Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows*
Dans cette tâche, vous installez et configurez IBM MQ sur un serveur ou un poste de travail dans le domaine *wmq.example.com* Windows.

Dans une configuration de niveau production, il peut s'avérer nécessaire de l'ajuster en fonction d'un domaine existant. Par exemple, vous pouvez être amené à définir des groupes de domaines différents afin d'autoriser différents partages et de grouper les ID utilisateur qui exécutent les gestionnaires de files d'attente.

La configuration exemple est constituée de trois serveurs :

sun

Un contrôleur de domaine Windows Server 2008. Il possède le domaine *wmq.example.com* qui contient *Sun, mars et venus*. A des fins de démonstration, il est également utilisé comme serveur de fichiers.

mars

Un serveur Windows Server 2008 utilisé en tant que premier serveur IBM MQ. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

venus

Un serveur Windows Server 2008 utilisé en tant que deuxième serveur IBM MQ. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

Important : Par défaut, le fonctionnement des ordinateurs avec Windows 10 version 1607 et Windows Server 2016 devient plus restrictifs qu'avec les versions antérieures de Windows.

Ce changement restreint les clients autorisés à effectuer des appels distants au gestionnaire de comptes de sécurité (SAM) et peut avoir un impact sur IBM MQ, notamment en empêchant les gestionnaires de files d'attente de démarrer. L'accès à SAM est essentiel pour le fonctionnement de IBM MQ lorsque IBM MQ est configuré en tant que compte de domaine.

1. Effectuez les étapes de la section «[Création d'un domaine Active Directory et DNS sur Windows](#)», à la page 556 pour créer un contrôleur de domaine, *sun*, pour le domaine *wmq.example.com*. Modifiez les noms en italiques afin de les adapter à votre configuration.
2. Voir [Configurations matérielles et logicielles sous Windows](#) pour connaître les autres versions Windows à partir desquelles vous pouvez exécuter IBM MQ.

Pourquoi et quand exécuter cette tâche

Dans cette tâche, vous configurez un Windows Server 2008, appelé *mars*, en tant que membre du domaine *wmq.example.com*. Vous installez IBM MQ et configurez l'installation pour qu'elle s'exécute en tant que membre du domaine *wmq.example.com*.

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix. Elles accompagnent la tâche «Domaines et gestionnaires de files d'attente multi-instances Windows», à la page 551.

Pour les besoins de la tâche, le nom d'hôte du contrôleur de domaine est *sun* et les deux serveurs IBM MQ sont appelés *mars* et *venus*. Le domaine est appelé *wmq.example.com*. Vous pouvez remplacer tous les noms en italique dans la tâche par les noms de votre choix.

Procédure

1. Ajoutez le contrôleur de domaine, *sun.wmq.example.com*, sur *mars* en tant que serveur DNS.
 - a) Sur *mars*, connectez-vous en tant que *mars\Administrator* et cliquez sur **Démarrer**.
 - b) Cliquez avec le bouton droit de la souris sur **Réseau > Propriétés > Gérer les connexions réseau**.
 - c) Cliquez avec le bouton droit sur l'adaptateur de réseau, puis sélectionnez **Propriétés**.

Le système répond en affichant la fenêtre Local Area Connection Properties qui répertorie les éléments utilisés par la connexion.
 - d) Sélectionnez **Internet Protocol Version 4** ou **Internet Protocol IBM WebSphere MQ 6** dans la liste des éléments de la fenêtre Propriétés de connexion à la zone locale. Cliquez sur **Propriétés > Avancé ...** et cliquez sur l'onglet **DNS**.
 - e) Sous les adresses de serveur DNS, cliquez sur **Add...**
 - f) Entrez l'adresse IP du contrôleur de domaine, qui est également le serveur DNS, et cliquez sur **Add**.
 - g) Cliquez sur **Ajouter ces suffixes DNS > Ajouter ...**
 - h) Entrez *wmq.example.com* et cliquez sur **Add**.
 - i) Entrez *wmq.example.com* dans la zone **DNS suffix for this connection**.
 - j) Sélectionnez **Register this connection's address in DNS** et **Use this connection's suffix in DNS registration**. Cliquez sur **OK > OK > Fermer**
 - k) Ouvrez une fenêtre de commande et entrez la commande **ipconfig /all** pour vérifier les paramètres TCP/IP.
2. Sur *mars*, ajoutez l'ordinateur au domaine *wmq.example.com*.
 - a) Cliquez sur **Démarrer**
 - b) Cliquez avec le bouton droit de la souris sur **Ordinateur > Propriétés**. Dans la zone Computer name, domain and workgroup settings, cliquez sur **Change settings**.
 - c) Dans les fenêtres System Properties, cliquez sur **Change...**
 - d) Cliquez sur Domain, entrez *wmq.example.com*, et cliquez sur **OK**.
 - e) Entrez le **Nom d'utilisateur** et le **Mot de passe** de l'administrateur du contrôleur de domaine, qui a le droit d'autoriser l'ordinateur à rejoindre le domaine et cliquez sur **OK**.
 - f) Cliquez sur **OK > OK > Fermer > Redémarrer maintenant** en réponse au message "Bienvenue dans le domaine *wmq.example.com*".
3. Vérifiez que l'ordinateur est membre du domaine *wmq.example.com*
 - a) Sur *sun*, connectez-vous au contrôleur de domaine en tant que *wmq\Administrator*.
 - b) Ouvrez **Server Manager > Active Directory Domain Services > wmq.example.com > Computers** et vérifiez que *mars* est correctement répertorié dans la fenêtre Computers.
4. Installez IBM MQ for Windows sur *mars*.

Pour en savoir plus sur la procédure d'exécution de l'assistant d'installation de IBM MQ for Windows, voir [Installation du serveur IBM MQ sous Windows](#).

- a) Sur *mars*, connectez-vous en tant qu'administrateur local, *mars\Administrator*.
- b) Exécutez la commande **Setup** sur le support d'installation de IBM MQ for Windows.
L'application de tableau de bord IBM MQ démarre.
- c) Cliquez sur **Software Requirements** pour vérifier que les logiciels prérequis sont installés.
- d) Cliquez sur **Configuration du réseau > Oui** pour configurer un ID utilisateur de domaine.
La tâche «Création d'un domaine Active Directory et DNS sur Windows», à la page 556 configure un ID utilisateur de domaine pour cet ensemble de tâches.
- e) Cliquez sur **IBM MQ Installation**, sélectionnez une langue d'installation et cliquez sur Lancer IBM MQ Installer.
- f) Confirmez le contrat de licence et cliquez sur **Suivant > Suivant > Installer** pour accepter la configuration par défaut. Attendez que l'installation se termine, puis cliquez sur **Finish**.

Vous pouvez si vous le souhaitez modifier le nom de l'installation, installer des composants différents, configurer un répertoire différent pour les données et les journaux du gestionnaire de files d'attente, ou encore effectuer l'installation dans un répertoire différent. Si tel est le cas, cliquez sur **Custom** au lieu de **Standard**.

IBM MQ est installé et le programme d'installation démarre Prepare IBM MQ Wizard.

Important : Ne lancez pas encore l'assistant.

5. Configurez l'utilisateur qui va exécuter le service IBM MQ avec le droit **Exécuter en tant que service** .

Choisissez de configurer le groupe *mqm* local, le groupe *Domain\mqm* ou l'utilisateur qui va exécuter le service IBM MQ avec le droit. Dans cet exemple, vous attribuez ce droit à l'utilisateur.

- a) Cliquez sur **Démarrer > Exécuter ...**, Entrez la commande **secpol.msc** et cliquez sur **OK**.
- b) Ouvrez **Paramètres de sécurité > Stratégies locales > Attribution des droits utilisateurs**. Dans la liste des règles, cliquez avec le bouton droit de la souris sur **Ouvrir une session en tant que service > Propriétés**.
- c) Cliquez sur **Ajouter un utilisateur ou un groupe...** et tapez *wmquser1* et cliquez sur **Vérifier les noms**
- d) Entrez le nom d'utilisateur et le mot de passe d'un administrateur de domaine, *wmq\Administrator*, puis cliquez sur **OK > Appliquer > OK**. Fermez la fenêtre Stratégie de sécurité locale.

6. Exécute Prepare IBM MQ Wizard.

Pour plus d'informations, voir [Configuration de IBM MQ avec Prepare IBM MQ Wizard](#).

- a) Le programme d'installation IBM MQ exécute le Prepare IBM MQ Wizard automatiquement.
Pour démarrer l'assistant manuellement, recherchez le raccourci vers Prepare IBM MQ Wizard dans le dossier **Démarrer > Tous les programmes > IBM MQ** . Sélectionnez le raccourci qui correspond à l'installation de IBM MQ dans une configuration multi-installation.
- b) Cliquez sur **Suivant** et laissez l'option **Oui** sélectionnée en réponse à la question "Identifier s'il existe un contrôleur de domaine Windows 2000 ou ultérieur dans le réseau".
- c) Cliquez sur **Oui > Suivant** dans la première fenêtre Configuration de IBM MQ for Windows pour les utilisateurs de domaine Windows .
- d) Dans la deuxième fenêtre Configuration de IBM MQ for Windows pour les utilisateurs de domaine Windows , entrez *wmq* dans la zone **Domaine** . Entrez *wmquser1* dans la zone **Nom d'utilisateur** et le mot de passe, le cas échéant, dans la zone **Mot de passe** . Cliquez sur **SUIVANT**.
L'assistant configure et démarre IBM MQ avec *wmquser1*.
- e) Dans la dernière page de l'assistant, sélectionnez ou désélectionnez les cases comme il convient et cliquez sur **Terminer**.

Que faire ensuite

1. Effectuez la tâche [«Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm»](#), à la page 585 pour vérifier que l'installation et la configuration fonctionnent correctement.
2. Réalisez la tâche [«Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows»](#), à la page 563 pour configurer un partage de fichiers pour le stockage des données et des fichiers journaux d'un gestionnaire de files d'attente multi-instance.

Tâches associées

Windows [Création d'un domaine Active Directory et DNS sur Windows](#)

Windows [Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows](#)

Windows [Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale](#)

Référence associée

[Droits utilisateur requis pour un service IBM MQ Windows](#)

Windows *Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows*

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix.

Dans une configuration de niveau production, il peut s'avérer nécessaire de l'ajuster en fonction d'un domaine existant. Par exemple, vous pouvez être amené à définir des groupes de domaines différents afin d'autoriser différents partages et de grouper les ID utilisateur qui exécutent les gestionnaires de files d'attente.

La configuration exemple est constituée de trois serveurs :

sun

Un contrôleur de domaine Windows Server 2008. Il possède le domaine *wmq.example.com* qui contient *Sun, mars et venus*. A des fins de démonstration, il est également utilisé comme serveur de fichiers.

mars

Un serveur Windows Server 2008 utilisé en tant que premier serveur IBM MQ. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

venus

Un serveur Windows Server 2008 utilisé en tant que deuxième serveur IBM MQ. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

1. Pour effectuer cette tâche exactement comme indiqué, effectuez les étapes de la tâche [«Création d'un domaine Active Directory et DNS sur Windows»](#), à la page 556 pour créer le domaine *sun.wmq.example.com* sur le contrôleur de domaine *sun*. Modifiez les noms en italiques afin de les adapter à votre configuration.

Pourquoi et quand exécuter cette tâche

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de

votre choix. Elles accompagnent la tâche «Domaines et gestionnaires de files d'attente multi-instances Windows», à la page 551.

Dans cette tâche, vous créez un partage contenant un répertoire de données et de journaux, ainsi qu'un groupe global pour autoriser l'accès au partage. Vous transmettez le nom du groupe global qui autorise le partage à la commande **crtmqm** dans son paramètre **-a**. Le groupe global vous offre la flexibilité de séparer les utilisateurs de ce partage des utilisateurs d'autres partages. Si vous n'avez pas besoin de cette flexibilité, autorisez le partage avec le groupe `Domain\mqm` plutôt que de créer un nouveau groupe global.

Le groupe global utilisé pour le partage dans cette tâche est appelé *wmqhaet* le partage est appelé *wmq*. Ils sont définis sur le contrôleur de domaine *sun* dans le Windows domaine *wmq.example.com*. Le partage dispose de droits de contrôle complets pour le groupe global *wmqha*. Remplacez les noms en italique dans la tâche par les noms de votre choix.

Pour les besoins de cette tâche, le contrôleur de domaine est le même serveur que le serveur de fichiers. Dans les applications pratiques, divisez les services de répertoire et de fichier entre les différents serveurs pour des raisons de performances et de disponibilité.

Vous devez configurer l'ID utilisateur sous lequel s'exécute le gestionnaire de files d'attente pour qu'il soit membre de deux groupes. Il doit être membre du groupe `mqm` local sur un serveur IBM MQ et du groupe global *wmqha*.

Dans cet ensemble de tâches, lorsque le gestionnaire de files d'attente s'exécute en tant que service, il s'exécute sous l'ID utilisateur *wmquser1*, de sorte que *wmquser1* doit être membre de *wmqha*. Lorsque le gestionnaire de files d'attente s'exécute en mode interactif, il s'exécute sous l'ID utilisateur *wmquser2*. Par conséquent, *wmquser2* doit être membre de *wmqha*. *wmquser1* et *wmquser2* sont tous deux membres du groupe global `Domain\mqm`. `Domain\mqm` est un membre du groupe `mqm` local sur les serveurs *mars* et *venus* IBM MQ. Par conséquent, *wmquser1* et *wmquser2* sont membres du groupe `mqm` local sur les deux serveurs IBM MQ.

Procédure

1. Connectez-vous au contrôleur de domaine, *sun.wmq.example.com* en tant qu'administrateur de domaine.
2. Créez le groupe global *wmqha*.
 - a) Ouvrez **Server Manager > Rôles > Active Directory Services de domaine > wmq.example.com > Utilisateurs**.
 - b) Ouvrez le dossier *wmq.example.com\Users*
 - c) Cliquez avec le bouton droit de la souris sur **Utilisateurs > Nouveau > Groupe**.
 - d) Entrez *wmqha* dans la zone **Nom du groupe**.
 - e) Laissez **Global** sélectionné comme **Portée du groupe** et **Sécurité** comme **Type de groupe**. Cliquez sur **OK**.
3. Ajoutez les utilisateurs de domaine *wmquser1* et *wmquser2* au groupe global, *wmqha*.
 - a) Dans l'arborescence de navigation du gestionnaire de serveur, cliquez sur **Utilisateurs** et cliquez avec le bouton droit de la souris sur **wmqha > Propriétés** dans la liste des utilisateurs.
 - b) Cliquez sur l'onglet Membres dans la fenêtre Propriétés *wmqha*.
 - c) Cliquez sur **Ajouter ...**; Tapez *wmquser1* ; *wmquser2* et cliquez sur **Vérifier les noms > OK > Appliquer > OK**.
4. Créez l'arborescence de répertoires pour contenir les données et les fichiers journaux du gestionnaire de files d'attente.
 - a) Ouvrez une invite de commande.
 - b) Tapez la commande :

```
md c:\wmq\data, c:\wmq\logs
```

5. Autorisez le groupe global *wmqha* à disposer de droits de contrôle complets sur les répertoires et le partage *c: \wmq* .
 - a) Dans Windows Explorer, cliquez avec le bouton droit de la souris sur **c: \wmq > Propriétés**.
 - b) Cliquez sur l'onglet **Sécurité** , puis sur **Avancé > Editer ...**
 - c) Décochez la case **Inclure les droits héritables du propriétaire de cet objet**. Cliquez sur **Copier** dans la fenêtre Windows Security.
 - d) Sélectionnez les lignes correspondant aux utilisateurs dans la liste des **entrées de droits d'accès** et cliquez sur **Supprimer**. Laissez les lignes pour SYSTEM, Administrators et CREATOR OWNER dans la liste des **entrées de droits d'accès**.
 - e) Cliquez sur **Ajouter ...**, et entrez le nom du groupe global *wmqha*. Cliquez sur **Vérifier les noms > OK**.
 - f) Dans la fenêtre Droits d'accès pour *wmq* , sélectionnez **Contrôle total** dans la liste **Droits d'accès**.
 - g) Cliquez sur **OK > Appliquer > OK > OK > OK**
 - h) Dans Windows Explorer, cliquez avec le bouton droit de la souris sur **c: \wmq > Partager ...**
 - i) Cliquez sur **Partage avancé ...** et cochez la case **Partager ce dossier** . Conservez le nom de partage *wmq*.
 - j) Cliquez sur **Droits > Ajouter ...**, et entrez le nom du groupe global *wmqha*. Cliquez sur **Vérifier les noms > OK**.
 - k) Sélectionnez *wmqha* dans la liste **Noms de groupe ou d'utilisateur**. Cochez la case **Contrôle total** dans la liste **Droits d'accès pour wmqha** ; cliquez sur **Appliquer**.
 - l) Sélectionnez *Administrators* dans la liste **Noms de groupe ou d'utilisateur**. Cochez la case **Contrôle total** dans la liste **Droits d'accès pour les administrateurs** ; cliquez sur **Appliquer > OK > OK > Fermer**.

Que faire ensuite

Vérifiez que vous pouvez lire et écrire des fichiers dans les répertoires partagés à partir de chacun des serveurs IBM MQ . Vérifiez l'ID utilisateur du service IBM MQ , *wmquser1* et l'ID utilisateur interactif, *wmquser2*.

1. Si vous utilisez le bureau à distance, vous devez ajouter *wmq\wmquser1* et *wmquser2* au groupe local Remote Desktop Users sur *mars*.
 - a. Connectez-vous à *mars* en tant que *wmq\Administrator*
 - b. Exécutez la commande **lusrmgr.msc** pour ouvrir la fenêtre Utilisateurs et groupes locaux.
 - c. Cliquez sur **Groupes**. Cliquez avec le bouton droit de la souris sur **Utilisateurs du bureau à distance > Propriétés > Ajouter ...**. Entrez *wmquser1* ; *wmquser2* et cliquez sur **Vérifier les noms**.
 - d. Entrez le nom d'utilisateur et le mot de passe de l'administrateur de domaine, *wmq\Administrator*, puis cliquez sur **OK > Appliquer > OK**.
 - e. Fermez la fenêtre Utilisateurs et groupes locaux.
2. Connectez-vous à *mars* en tant que *wmq\wmquser1*.
 - a. Ouvrez une fenêtre Windows Explorer et entrez `\\sun\wmq`.
Le système répond en ouvrant le partage *wmq* sur *sun.wmq.example.comet* en répertoriant les répertoires de données et de journaux.
 - b. Vérifiez les droits d'accès de *wmquser1* en créant un fichier dans un sous-répertoire de données, en ajoutant du contenu, en le lisant, puis en le supprimant.
3. Connectez-vous à *mars* en tant que *wmq\wmquser2* et répétez les vérifications.
4. Effectuez la tâche suivante pour créer un gestionnaire de files d'attente afin d'utiliser les répertoires de données et de journaux partagés ; voir «[Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale](#)», à la page 566.

Tâches associées

Windows [Création d'un domaine Active Directory et DNS sur Windows](#)

Windows [Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows](#)

Windows [Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale](#)

Windows [Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale](#)

Cette tâche explique comment utiliser l'indicateur -a dans la commande `crtmqm`. L'indicateur -a permet au gestionnaire de files d'attente d'accéder à ses fichiers journaux et de données sur un partage de fichiers distant à l'aide du groupe de sécurité alternatif.

Dans une configuration de niveau production, il peut s'avérer nécessaire de l'ajuster en fonction d'un domaine existant. Par exemple, vous pouvez être amené à définir des groupes de domaines différents afin d'autoriser différents partages et de grouper les ID utilisateur qui exécutent les gestionnaires de files d'attente.

La configuration exemple est constituée de trois serveurs :

sun

Un contrôleur de domaine Windows Server 2008. Il possède le domaine *wmq.example.com* qui contient *Sun, mars et venus*. A des fins de démonstration, il est également utilisé comme serveur de fichiers.

mars

Un serveur Windows Server 2008 utilisé en tant que premier serveur IBM MQ. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

venus

Un serveur Windows Server 2008 utilisé en tant que deuxième serveur IBM MQ. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

Effectuez les étapes des tâches suivantes. Les tâches créent le contrôleur de domaine et le domaine, installent IBM MQ for Windows sur un serveur et créent le partage de fichiers pour les données et les fichiers journaux. Si vous configurez un contrôleur de domaine existant, il peut s'avérer utile de tester les étapes sur un nouveau serveur Windows 2008. Vous pouvez adapter les étapes à votre domaine.

1. [«Création d'un domaine Active Directory et DNS sur Windows»](#), à la page 556.
2. [«Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows»](#), à la page 560.
3. [«Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows»](#), à la page 563.

Pourquoi et quand exécuter cette tâche

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix. Elles accompagnent la tâche [«Domaines et gestionnaires de files d'attente multi-instances Windows»](#), à la page 551.

Dans cette tâche, vous créez un gestionnaire de files d'attente qui stocke ses données et ses journaux dans un répertoire distant sur un serveur de fichiers. Pour les besoins de cet exemple, le serveur de fichiers est le même que le contrôleur de domaine. Le répertoire contenant les dossiers de données et de journaux est partagé avec les droits de contrôle complets accordés au groupe global *wmqha*.

Procédure

1. Connectez-vous au serveur de domaine, *mars*, en tant qu'administrateur local, *mars\Administrator*.
2. Ouvrez une fenêtre de commande.
3. Redémarrez le service IBM MQ.

Vous devez redémarrer le service pour que l'ID utilisateur sous lequel il s'exécute acquière les données d'identification de sécurité supplémentaires que vous avez configurées pour lui.

Entrez les commandes suivantes:

```
endmqsvc  
strmqsvc
```

Les réponses du système:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
The MQ service for installation 'Installation1' ended successfully.
```

Et:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
The MQ service for installation 'Installation1' started successfully.
```

4. Créez le gestionnaire de files d'attente.

```
crtmqm -a wmq\wmqha -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\sun\wmq\data -ld \\sun\wmq\logs  
QMGR
```

Vous devez spécifier le domaine, *wmq*, du groupe de sécurité alternatif *wmqha* en spécifiant le nom de domaine complet du groupe global "*wmq\wmqha*".

Vous devez définir le nom UNC (Universal Naming Convention) du partage *\\sun\wmqet* ne pas utiliser de référence d'unité mappée.

Réponse du système :

```
IBM MQ queue manager created.  
Directory '\\sun\wmq\data\QMGR' created.  
The queue manager is associated with installation '1'  
Creating or replacing default objects for queue manager 'QMGR'  
Default objects statistics : 74 created. 0 replaced.  
Completing setup.  
Setup completed.
```

Que faire ensuite

Testez le gestionnaire de files d'attente en insérant un message dans une file d'attente, puis en recevant un message.

1. Démarrez le gestionnaire de files d'attente.

```
strmqm QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.
```

The queue manager is associated with installation '1'.
5 log records accessed on queue manager 'QMGR' during the log replay phase.
Log replay for queue manager 'QMGR' complete.
Transaction manager state recovered for queue manager 'QMGR'.
IBM MQ queue manager 'QMGR' started using V7.1.0.0.

2. Créez une file d'attente de test.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Réponse du système :

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)  
AMQ8006: IBM MQ queue created.  
One MQSC command read.  
No commands have a syntax error.  
All valid MQSC commands were processed.
```

3. Insérez un message de test à l'aide de l'exemple de programme **amqsput**.

```
echo 'A test message' | amqsput QTEST QMGR
```

Réponse du système :

```
Sample AMQSPUT0 start  
target queue is QTEST  
Sample AMQSPUT0 end
```

4. Recevez le message de test à l'aide de l'exemple de programme **amqsget**.

```
amqsget QTEST QMGR
```

Réponse du système :

```
Sample AMQSGET0 start  
message A test message  
Wait 15 seconds ...  
no more messages  
Sample AMQSGET0 end
```

5. Arrêtez le gestionnaire de files d'attente.

```
endmqm -i QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ended.
```

6. Supprimez le gestionnaire de files d'attente.


```
dltmqm QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' deleted.
```

7. Supprimez les répertoires que vous avez créés.

Conseil : Ajoutez l'option /Q aux commandes pour empêcher la commande invitant à supprimer chaque fichier ou répertoire.

```
del /F /S C:\wmq\*. *
rmdir /S C:\wmq
```

Tâches associées

Windows [Création d'un domaine Active Directory et DNS sur Windows](#)

Windows [Installation de IBM MQ sur un serveur ou un poste de travail dans un domaine Windows](#)

Windows [Création d'un répertoire partagé pour les données du gestionnaire de files d'attente et les fichiers journaux sous Windows](#)

Windows [Création d'un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine Windows](#)

Un exemple montre comment configurer un gestionnaire de files d'attente multi-instance sur Windows sur des contrôleurs de domaine. La configuration illustre les concepts concernés plutôt que de traiter le niveau de production. L'exemple est basé sur Windows Server 2008. Les étapes peuvent varier d'une version à l'autre de Windows Server.

La configuration utilise le concept de mini-domaine, ou "domainlet" ; voir [Windows 2000, Windows Server 2003 et Windows Server 2008 cluster nodes en tant que contrôleurs de domaine](#). Pour ajouter des gestionnaires de files d'attente multi-instance à un domaine existant, voir «[Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows](#)», à la page 553.

La configuration exemple est constituée de trois serveurs :

sun

Un serveur Windows Server 2008 utilisé comme premier contrôleur de domaine. Il définit le domaine *wmq.example.com* qui contient *sun*, *earth* et *mars*. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

earth

Un serveur Windows Server 2008 utilisé comme deuxième serveur IBM MQ de contrôleur de domaine. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

mars

Un serveur Windows Server 2008 utilisé comme serveur de fichiers.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

1. Sous Windows, vous n'avez pas besoin de vérifier le système de fichiers sur lequel vous prévoyez de stocker les données et les fichiers journaux du gestionnaire de files d'attente. La procédure de vérification [Vérification du comportement du système de fichiers partagé](#), s'applique à AIX and Linux. Sous Windows, les vérifications aboutissent toujours.
2. Effectuez les étapes de la rubrique «[Création d'un domaine Active Directory et DNS sur Windows](#)», à la page 556 pour créer le premier contrôleur de domaine.

3. Effectuez les étapes de la rubrique «Ajout d'un second contrôleur de domaine Windows à un exemple de domaine», à la page 573 pour ajouter un second contrôleur de domaine, installer IBM MQ for Windows sur les deux contrôleurs de domaine et vérifier les installations.
4. Effectuez les étapes de la rubrique «Installation de IBM MQ sur des contrôleurs de domaine Windows dans un exemple de domaine», à la page 575 pour installer IBM MQ sur les deux contrôleurs de domaine.

Pourquoi et quand exécuter cette tâche

Sur un serveur de fichiers du même domaine, créez un partage pour le journal du gestionnaire de files d'attente et les répertoires de données. Ensuite, créez la première instance d'un gestionnaire de files d'attente multi-instance qui utilise le partage de fichiers sur l'un des contrôleurs de domaine. Créez l'autre instance sur l'autre contrôleur de domaine et vérifiez la configuration. Vous pouvez créer le partage de fichiers sur un contrôleur de domaine.

Dans l'exemple, *sun* est le premier contrôleur de domaine, *earth* le second et *mars* le serveur de fichiers.

Procédure

1. Créez les répertoires qui doivent contenir les données du gestionnaire de files d'attente et les fichiers journaux.
 - a) Sous *mars*, entrez la commande suivante:

```
md c:\wmq\data , c:\wmq\logs
```

2. Partagez les répertoires qui doivent contenir les données et les fichiers journaux du gestionnaire de files d'attente.

Vous devez autoriser un accès de contrôle total au groupe local de domaine *mqmet* à l'ID utilisateur que vous utilisez pour créer le gestionnaire de files d'attente. Dans l'exemple, les ID utilisateur membres de *Domain Administrators* sont autorisés à créer des gestionnaires de files d'attente.

Le partage de fichiers doit se trouver sur un serveur qui se trouve dans le même domaine que les contrôleurs de domaine. Dans l'exemple, le serveur *mars* se trouve dans le même domaine que les contrôleurs de domaine.

- a) Dans Windows Explorer, cliquez avec le bouton droit de la souris sur **c: \wmq > Propriétés**.
- b) Cliquez sur l'onglet **Sécurité**, puis sur **Avancé > Editer ...**.
- c) Décochez la case **Inclure les droits héritables du propriétaire de cet objet**. Cliquez sur **Copier** dans la fenêtre Windows Security.
- d) Sélectionnez les lignes correspondant aux utilisateurs dans la liste des **entrées de droits d'accès** et cliquez sur **Supprimer**. Laissez les lignes pour SYSTEM, Administrators et CREATOR OWNER dans la liste des **entrées de droits d'accès**.
- e) Cliquez sur **Ajouter ...**, et entrez le nom du groupe local de domaine *mqm*. Cliquez sur **Propriétés**.
- f) En réponse à une fenêtre de sécurité Windows, entrez le nom et le mot de passe du *Domain Administrator* et cliquez sur **OK > OK**.
- g) Dans la fenêtre Droits d'accès pour *wmq*, sélectionnez **Contrôle total** dans la liste **Droits d'accès**.
- h) Cliquez sur **OK > Appliquer > OK > OK > OK**.
- i) Répétez les étapes **e** à **h** pour ajouter *Domain Administrators*.
- j) Dans Windows Explorer, cliquez avec le bouton droit de la souris sur **c: \wmq > Partager ...**.
- k) Cliquez sur **Partage avancé ...** et cochez la case **Partager ce dossier**. Conservez le nom de partage *wmq*.
- l) Cliquez sur **Droits > Ajouter ...**, et entrez le nom du groupe local de domaine *mqm* ; *Domain Administrators*. Cliquez ensuite sur **Propriétés**.

m) En réponse à une fenêtre de sécurité Windows , entrez le nom et le mot de passe du Domain Administrator et cliquez sur **OK > OK**.

3. Créez le gestionnaire de files d'attente *QMGR* sur le premier contrôleur de domaine, *sun*.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md \\mars\wmq\data -ld \\mars\wmq\logs QMGR
```

Réponse du système :

```
IBM MQ queue manager created.  
Directory '\\mars\wmq\data\QMGR' created.  
The queue manager is associated with installation 'Installation1'.  
Creating or replacing default objects for queue manager 'QMGR'.  
Default objects statistics : 74 created. 0 replaced. 0 failed.  
Completing setup.  
Setup completed.
```

4. Démarrez le gestionnaire de files d'attente sous *sun*, en autorisant une instance de secours.

```
strmqm -x QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

5. Créez une deuxième instance de *QMGR* sur *earth*.

a) Vérifiez que les valeurs des paramètres Prefix et InstallationName sont correctes pour *earth*.

Sous *sun*, exécutez la commande **dspmqinf** :

```
dspmqinf QMGR
```

Réponse du système :

```
QueueManager:  
Name=QMGR  
Directory=QMGR  
Prefix=C:\ProgramData\IBM\MQ  
DataPath=\\mars\wmq\data\QMGR  
InstallationName=Installation1
```

b) Copiez le formulaire lisible par la machine de la section **QueueManager** vers le presse-papiers.

Sous *sun* , exécutez à nouveau la commande **dspmqinf** avec le paramètre -o command .

```
dspmqinf -o command QMGR
```

Réponse du système :

```
addmqinf -s QueueManager -v Name=QMGR
```

```
-v Directory=QMGR -v Prefix="C:\ProgramData\IBM\MQ"  
-v DataPath=\\mars\wmq\data\QMGR
```

- c) Sous *earth*, exécutez la commande **addmqinf** à partir du presse-papiers pour créer une instance du gestionnaire de files d'attente sous *earth*.

Si nécessaire, ajustez la commande en fonction des différences dans les paramètres Prefix ou InstallationName.

```
addmqinf -s QueueManager -v Name= QMGR  
-v Directory= QMGR -v Prefix="C:\Program Files\IBM\WebSphere MQ"  
-v DataPath=\\mars\wmq\data\QMGR
```

IBM MQ configuration information added.

6. Démarrez l'instance de secours du gestionnaire de files d'attente sous *earth*.

```
strmqm -x QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation 'Installation1'.  
A standby instance of queue manager 'QMGR' has been started. The active  
instance is running elsewhere.
```

Résultats

Vérifiez que le gestionnaire de files d'attente bascule de *sun* vers *earth*:

1. Sous *sun*, exécutez la commande suivante:

```
endmqm -i -r -s QMGR
```

Réponse du système sous *sun*:

```
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ended, permitting switchover to  
a standby instance.
```

2. Sous *earth*, entrez à plusieurs reprises la commande suivante:

```
dspmqr
```

Les réponses du système:

```
QMNAME(QMGR) STATUS(Running as standby)  
QMNAME(QMGR) STATUS(Running as standby)  
QMNAME(QMGR) STATUS(Running)
```

Que faire ensuite

Pour vérifier un gestionnaire de files d'attente multi-instance par le biais de plusieurs exemples de programme, voir [«Vérification du gestionnaire de files d'attente multi-instance sous Windows»](#), à la page 577.

Tâches associées

[«Ajout d'un second contrôleur de domaine Windows à un exemple de domaine»](#), à la page 573

[«Installation de IBM MQ sur des contrôleurs de domaine Windows dans un exemple de domaine»](#), à la page 575

Information associée

[Noeuds de cluster Windows 2000, Windows Server 2003 et Windows Server 2008 en tant que contrôleurs de domaine](#)

Windows Ajout d'un second contrôleur de domaine Windows à un exemple de domaine

Ajoutez un deuxième contrôleur de domaine au domaine *wmq.example.com* pour construire un domaine Windows dans lequel exécuter des gestionnaires de files d'attente multi-instance sur des contrôleurs de domaine et des serveurs de fichiers.

La configuration exemple est constituée de trois serveurs :

sun

Un serveur Windows Server 2008 utilisé comme premier contrôleur de domaine. Il définit le domaine *wmq.example.com* qui contient *sun*, *earth* et *mars*. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

earth

Un serveur Windows Server 2008 utilisé comme deuxième serveur IBM MQ de contrôleur de domaine. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

mars

Un serveur Windows Server 2008 utilisé comme serveur de fichiers.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

1. Effectuez les étapes de la section [«Création d'un domaine Active Directory et DNS sur Windows»](#), à la page 556 pour créer un contrôleur de domaine, *sun*, pour le domaine *wmq.example.com*. Modifiez les noms en italiques afin de les adapter à votre configuration.
2. Installez Windows Server 2008 sur un serveur du groupe de travail par défaut, WORKGROUP. Pour l'exemple, le serveur est nommé *earth*.

Pourquoi et quand exécuter cette tâche

Dans cette tâche, vous configurez un serveur Windows 2008, appelé *earth*, en tant que deuxième contrôleur de domaine dans le domaine *wmq.example.com*.

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix. Elles accompagnent la tâche [«Domaines et gestionnaires de files d'attente multi-instances Windows»](#), à la page 551.

Procédure

1. Ajoutez le contrôleur de domaine *sun.wmq.example.com* à *earth* en tant que serveur DNS.
 - a) Sous *earth*, connectez-vous en tant que *earth\Administrator* et cliquez sur **Démarrer**.
 - b) Cliquez avec le bouton droit de la souris sur **Réseau > Propriétés > Gérer les connexions réseau**.
 - c) Cliquez avec le bouton droit sur l'adaptateur de réseau, puis sélectionnez **Properties**.

Le système répond en affichant la fenêtre Local Area Connection Properties qui répertorie les éléments utilisés par la connexion.

- d) Sélectionnez **Internet Protocol Version 4** ou **Internet Protocol IBM WebSphere MQ 6** dans la liste des éléments de la fenêtre Propriétés de connexion à la zone locale. Cliquez sur **Propriétés > Avancé ...** et cliquez sur l'onglet **DNS**.
 - e) Sous les adresses de serveur DNS, cliquez sur **Add...**
 - f) Entrez l'adresse IP du contrôleur de domaine, qui est également le serveur DNS, et cliquez sur **Add**.
 - g) Cliquez sur **Ajouter ces suffixes DNS > Ajouter**
 - h) Entrez *wmq.example.com* et cliquez sur **Add**.
 - i) Entrez *wmq.example.com* dans la zone **DNS suffix for this connection**.
 - j) Sélectionnez **Register this connection's address in DNS** et **Use this connection's suffix in DNS registration**. Cliquez sur **OK > OK > Fermer**
 - k) Ouvrez une fenêtre de commande et entrez la commande **ipconfig /all** pour vérifier les paramètres TCP/IP.
2. Connectez-vous au contrôleur de domaine, *sun*, en tant qu'administrateur local ou `Workgroup`.
Si le serveur est déjà configuré en tant que contrôleur de domaine, vous devez vous connecter en tant qu'administrateur de domaine.
 3. Exécutez l'assistant de service de domaine Active Directory.
 - a) Cliquez sur **Démarrer > Exécuter ...** Entrez `dcprromo` et cliquez sur **OK**.
Si les fichiers binaires Active Directory ne sont pas déjà installés, Windows installe les fichiers automatiquement.
 4. Configurez *earth* en tant que deuxième contrôleur de domaine dans le domaine *wmq.example.com*.
 - a) Dans la première fenêtre de l'assistant, désélectionnez la case **Use advanced mode installation**. Cliquez sur **Suivant > Suivant**, puis sur **Créer Ajouter un contrôleur de domaine à un domaine existant > Suivant**.
 - b) Entrez *wmq* dans la zone **Entrez le nom d'un domaine dans cette forêt ...**:NONE. Cliquez sur le bouton d'option **Données d'identification de remplacement** et cliquez sur **Définir** Entrez le nom et le mot de passe de l'administrateur de domaine et cliquez sur **OK > Suivant > Suivant > Suivant**.
 - c) Dans la fenêtre Options de contrôleur de domaine supplémentaires, acceptez les options **Serveur DNS** et **Catalogue global**, qui sont sélectionnées ; cliquez sur **Suivant > Suivant**.
 - d) Dans le mot de passe de l'administrateur du mode de restauration des services d'annuaire, entrez un **mot de passe** et **confirmez le mot de passe**, puis cliquez sur **Suivant > Suivant**.
 - e) A l'invite **Network Credentials**, entrez le mot de passe de l'administrateur de domaine. Sélectionnez **Reboot on completion** dans la fenêtre de l'assistant final.
 - f) Après un certain temps, une fenêtre peut s'ouvrir avec une erreur **DCPrromo** concernant la délégation DNS ; cliquez sur **OK**. Le serveur redémarre.


Résultats

Une fois *earth* redémarré, connectez-vous en tant qu'administrateur de domaine. Vérifiez que le domaine *wmq.example.com* a été répliqué sur *earth*.

Que faire ensuite

Poursuivez l'installation de IBM MQ ; voir [«Installation de IBM MQ sur des contrôleurs de domaine Windows dans un exemple de domaine»](#), à la page 575.

Tâches associées

 [Installation de IBM MQ sur des contrôleurs de domaine Windows dans un exemple de domaine](#)

[«Création d'un domaine Active Directory et DNS sur Windows», à la page 556](#)

Windows *Installation de IBM MQ sur des contrôleurs de domaine Windows dans un exemple de domaine*

Installez et configurez les installations d' IBM MQ sur les deux contrôleurs de domaine du domaine *wmq.example.com*.

La configuration exemple est constituée de trois serveurs :

sun

Un serveur Windows Server 2008 utilisé comme premier contrôleur de domaine. Il définit le domaine *wmq.example.com* qui contient *sun*, *earth* et *mars*. Il contient une instance du gestionnaire de files d'attente multi-instance appelée *QMGR*.

earth

Un serveur Windows Server 2008 utilisé comme deuxième serveur IBM MQ de contrôleur de domaine. Il contient la deuxième instance du gestionnaire de files d'attente multi-instance appelé *QMGR*.

mars

Un serveur Windows Server 2008 utilisé comme serveur de fichiers.

Remplacez les noms présentés en italique dans l'exemple par les noms de votre choix.

Avant de commencer

1. Effectuez les étapes de la section [«Création d'un domaine Active Directory et DNS sur Windows», à la page 556](#) pour créer un contrôleur de domaine, *sun*, pour le domaine *wmq.example.com*. Modifiez les noms en italiques afin de les adapter à votre configuration.
2. Effectuez les étapes de la rubrique [«Ajout d'un second contrôleur de domaine Windows à un exemple de domaine», à la page 573](#) pour créer un second contrôleur de domaine, *earth*, pour le domaine *wmq.example.com*. Modifiez les noms en italiques afin de les adapter à votre configuration.
3. Voir [Configurations matérielles et logicielles sous Windows](#) pour connaître les autres versions Windows à partir desquelles vous pouvez exécuter IBM MQ.

Pourquoi et quand exécuter cette tâche

Installez et configurez les installations d' IBM MQ sur les deux contrôleurs de domaine du domaine *wmq.example.com*.

Procédure

1. Installez IBM MQ sous *sun* et *earth*.

Pour plus d'informations, voir [Installation du serveur IBM MQ sur Windows](#).

- a) Sur *sun* et *earth*, connectez-vous en tant qu'administrateur de domaine, *wmq\Administrator*.
- b) Exécutez la commande **Setup** sur le support d'installation de IBM MQ for Windows.

L'application de tableau de bord IBM MQ démarre.

- c) Cliquez sur **Software Requirements** pour vérifier que les logiciels prérequis sont installés.
- d) Cliquez sur **Configuration de réseau > Non**.

Vous pouvez configurer un ID utilisateur de domaine ou non pour cette installation. L'ID utilisateur créé est un ID utilisateur local de domaine.

- e) Cliquez sur **IBM MQ Installation**, sélectionnez une langue d'installation et cliquez sur Lancer IBM MQ Installer.
- f) Confirmez le contrat de licence et cliquez sur **Suivant > Suivant > Installer** pour accepter la configuration par défaut. Attendez que l'installation se termine, puis cliquez sur **Finish**.

Si vous souhaitez modifier le nom de l'installation, installer des composants différents, configurer un répertoire différent pour les données et les journaux du gestionnaire de files d'attente ou effectuer l'installation dans un répertoire différent, cliquez sur **Personnalisé** au lieu de **Standard**.

IBM MQ est installé et le programme d'installation démarre Prepare IBM MQ Wizard.

L'installation IBM MQ for Windows configure un groupe local de domaine mqmet un groupe de domaine Domain mqm. Elle fait de Domain mqm un membre de mqm. Les contrôleurs de domaine suivants du même domaine partagent les groupes mqm et Domain mqm.

2. Sur *earth* et *sun*, exécutez Prepare IBM MQ Wizard.

Pour plus d'informations, voir [Configuration de IBM MQ avec Prepare IBM MQ Wizard](#).

- a) Le programme d'installation de IBM MQ exécute le Prepare IBM MQ Wizard automatiquement.

Pour démarrer l'assistant manuellement, recherchez le raccourci vers Prepare IBM MQ Wizard dans le dossier **Démarrer > Tous les programmes > IBM MQ**. Sélectionnez le raccourci qui correspond à l'installation de IBM MQ dans une configuration multi-installation.

- b) Cliquez sur **Suivant** et laissez l'option **Non** sélectionnée en réponse à la question "Identifier s'il existe un contrôleur de domaine Windows 2000 ou ultérieur dans le réseau"¹.
- c) Dans la dernière page de l'assistant, sélectionnez ou désélectionnez les cases comme il convient et cliquez sur **Terminer**.

Prepare IBM MQ Wizard crée un utilisateur local de domaine MUSR_MQADMIN sur le premier contrôleur de domaine et un autre utilisateur local de domaine MUSR_MQADMIN1 sur le deuxième contrôleur de domaine. L'assistant crée le service IBM MQ sur chaque contrôleur, avec MUSR_MQADMIN ou MUSR_MQADMIN1 comme utilisateur qui se connecte au service.

3. Définissez un utilisateur autorisé à créer un gestionnaire de files d'attente.

L'utilisateur doit avoir le droit de se connecter localement et être membre du groupe mqm local du domaine. Sur les contrôleurs de domaine, les utilisateurs de domaine n'ont pas le droit de se connecter en local, contrairement aux administrateurs. Par défaut, aucun utilisateur ne possède ces deux attributs. Dans cette tâche, ajoutez des administrateurs de domaine au groupe mqm local du domaine.

- a) Ouvrez **Server Manager > Rôles > Active Directory Services de domaine > wmq.example.com > Utilisateurs**.
- b) Cliquez avec le bouton droit de la souris sur **Domain Admins > Ajouter à un groupe ...** et entrez mqm ; cliquez sur **Vérifier les noms > OK > OK**

Résultats

1. Vérifiez que Prepare IBM MQ Wizard a créé l'utilisateur de domaine MUSR_MQADMIN:

- a. Ouvrez **Server Manager > Rôles > Active Directory Services de domaine > wmq.example.com > Utilisateurs**.
- b. Cliquez avec le bouton droit de la souris sur **MUSR_MQADMIN > Propriétés ... > Membre de**, et vérifiez qu'il est membre de Domain users et mqm.

2. Vérifiez que MUSR_MQADMIN a le droit de s'exécuter en tant que service:


- a. Cliquez sur Cliquez sur **Démarrer > Exécuter ...**, Entrez la commande **secpol.msc** et cliquez sur **OK**.
- b. Ouvrez **Paramètres de sécurité > Stratégies locales > Attribution des droits utilisateurs**. Dans la liste des règles, cliquez avec le bouton droit de la souris sur **Ouvrir une session en tant que service > Propriétés**, et voir MUSR_MQADMIN est répertorié comme ayant le droit de se connecter en tant que service. Cliquez sur **OK**.

¹ Vous pouvez configurer l'installation pour le domaine. Comme tous les utilisateurs et groupes d'un contrôleur de domaine ont une portée de domaine, cela ne fait aucune différence. Il est plus simple d'installer IBM MQ comme s'il ne se trouve pas dans le domaine.

Que faire ensuite

1. Effectuez la tâche «[Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm](#)», à la page 585 pour vérifier que l'installation et la configuration fonctionnent correctement.
2. Revenez à la tâche «[Création d'un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine Windows](#)», à la page 569 pour terminer la tâche de configuration d'un gestionnaire de files d'attente multi-instance sur les contrôleurs de domaine.

Tâches associées

 [Ajout d'un second contrôleur de domaine Windows à un exemple de domaine](#)

Référence associée

[Droits utilisateur requis pour un service IBM MQ Windows](#)

 [Vérification du gestionnaire de files d'attente multi-instance sous Windows](#)

Utilisez les exemples de programme **amqsgbac**, **amqspbac** et **amqsmbac** pour vérifier une configuration de gestionnaire de files d'attente multi-instance. Cette rubrique fournit un exemple de configuration permettant de vérifier une configuration de gestionnaire de files d'attente multi-instance sur Windows Server 2003.

Les exemples de programmes à haute disponibilité utilisent la reconnexion automatique du client. Lorsque le gestionnaire de files d'attente connecté échoue, le client tente de se reconnecter à un gestionnaire de files d'attente du même groupe de gestionnaires de files d'attente. La description des exemples, [Exemples de programmes à haute disponibilité](#), illustre la reconnexion du client à l'aide d'un gestionnaire de files d'attente à instance unique pour plus de simplicité. Vous pouvez utiliser les mêmes exemples avec des gestionnaires de files d'attente multi-instance pour vérifier une configuration de gestionnaire de files d'attente multi-instance.

Cet exemple utilise la configuration multi-instance décrite dans «[Création d'un gestionnaire de files d'attente multi-instance sur des contrôleurs de domaine Windows](#)», à la page 569. Utilisez la configuration pour vérifier que le gestionnaire de files d'attente multi-instance bascule sur l'instance de secours. Arrêtez le gestionnaire de files d'attente à l'aide de la commande **endmqm** et utilisez l'option **-s, switchover**. Les programmes client se reconnectent à la nouvelle instance de gestionnaire de files d'attente et continuent à utiliser la nouvelle instance après un léger délai.

Le client est installé dans une image VMware 400 Mo qui exécute Windows 7 Service Pack 1. Pour des raisons de sécurité, il est connecté sur le même réseau hôte VMware que les serveurs de domaine qui exécutent le gestionnaire de files d'attente multi-instance. Il partage le dossier /MQHA, qui contient la table de connexion client, pour simplifier la configuration.

Vérification de la reprise en ligne avec IBM MQ Explorer

Avant d'utiliser les modèles d'application pour vérifier la reprise en ligne, exécutez le IBM MQ Explorer sur chaque serveur. Ajoutez les deux instances de gestionnaire de files d'attente à chaque explorateur à l'aide de l'assistant **Ajout d'un gestionnaire de files d'attente distantes > Connexion directe à un gestionnaire de files d'attente multi-instance**. Vérifiez que les deux instances sont en cours d'exécution, en autorisant la mise en veille. Fermez la fenêtre exécutant l'image VMware avec l'instance active, mettez virtuellement le serveur hors tension ou arrêtez l'instance active, ce qui permet de basculer vers l'instance de secours et de reconnecter les clients.



Avertissement : Si vous mettez le serveur hors tension, assurez-vous qu'il ne s'agit pas de celui qui héberge le dossier MQHA !

Remarque : L'option **Autoriser le basculement vers une instance de secours** peut ne pas être disponible dans la boîte de dialogue **Arrêter un gestionnaire de files d'attente**. L'option est manquante car le gestionnaire de files d'attente s'exécute en tant que gestionnaire de files d'attente à instance unique. Vous devez l'avoir démarré sans l'option **Autoriser une instance de secours**. Si votre demande d'arrêt du gestionnaire de files d'attente est rejetée, consultez la fenêtre **Détails**, il est possible qu'aucune instance de secours ne soit en cours d'exécution.

Vérification de la reprise en ligne à l'aide des exemples de programme

Choisissez un serveur pour exécuter l'instance active

Vous avez peut-être choisi l'un des serveurs pour héberger le répertoire MQHA ou le système de fichiers. Si vous prévoyez de tester la reprise en ligne en fermant la fenêtre VMware qui exécute le serveur actif, assurez-vous qu'il ne s'agit pas de celui qui héberge MQHA !

Sur le serveur exécutant l'instance de gestionnaire de files d'attente active

1. Modifiez *ipaddr1* et *ipaddr2* et sauvegardez les commandes suivantes dans `N:\hasample.tst`.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER(' ') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNNAME(' ipaddr1 (1414), ipaddr2 (1414)') QMNAME(QM1) REPLACE
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

Remarque : Si vous ne renseignez pas le paramètre **MCAUSER**, l'ID utilisateur du client est envoyé au serveur. L'ID utilisateur client doit disposer des droits appropriés sur les serveurs. Vous pouvez également définir le paramètre **MCAUSER** dans le canal SVRCONN sur l'ID utilisateur que vous avez configuré sur le serveur.

2. Ouvrez une invite de commande avec le chemin `N:\` et exécutez la commande suivante:

```
runmqsc -m QM1 < hasample.tst
```

3. Vérifiez que le programme d'écoute est en cours d'exécution et qu'il a le contrôle du gestionnaire de files d'attente, en examinant la sortie de la commande **runmqsc**.

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

Ou bien, en utilisant le IBM MQ Explorer que le programme d'écoute TCPIP est en cours d'exécution et possède `Control = Queue Manager`.

Sur le client

1. Mappez le répertoire partagé `C:\MQHA` sur le serveur à `N:\` sur le client.
2. Ouvrez une invite de commande avec le chemin `N:\`. Définissez la variable d'environnement `MQCHLLIB` pour qu'elle pointe vers la table de définition de canal du client (CCDT) sur le serveur:

```
SET MQCHLLIB=N:\data\QM1\@ipcc
```

3. A l'invite de commande, entrez les commandes suivantes:

```
start amqsgnac TARGET QM1
start amqsmnac -s SOURCE -t TARGET -m QM1
start amqspnac SOURCE QM1
```

Remarque : Si vous rencontrez des problèmes, démarrez les applications à partir d'une invite de commande afin que le code anomalie soit imprimé sur la console, ou consultez `AMQERR01.LOG` dans le dossier `N:\data\QM1\errors`.

Sur le serveur exécutant l'instance de gestionnaire de files d'attente active

1. L'un ou l'autre :
 - Fermez la fenêtre exécutant l'image VMware avec l'instance de serveur active.

- A l'aide de la IBM MQ Explorer, arrêtez l'instance de gestionnaire de files d'attente active, en autorisant la commutation vers l'instance de secours et en demandant aux clients reconnectables de se reconnecter.
2. Les trois clients détectent finalement que la connexion est interrompue, puis se reconnectent. Dans cette configuration, si vous fermez la fenêtre du serveur, le rétablissement des trois connexions prend environ sept minutes. Certaines connexions sont rétablies bien avant d'autres.

Résultats

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsgnac TARGET QM1
Sample AMQSGHAC start
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

Windows *Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows*

Cette rubrique explique comment sécuriser un emplacement partagé pour les données du gestionnaire de files d'attente et les fichiers journaux à l'aide d'un autre groupe de sécurité global. Vous pouvez partager l'emplacement entre différentes instances d'un gestionnaire de files d'attente s'exécutant sur des serveurs différents.

En règle générale, vous ne configurez pas d'emplacement partagé pour les données et les fichiers journaux du gestionnaire de files d'attente. Lorsque vous installez IBM MQ for Windows, le programme d'installation crée un répertoire de base de votre choix pour tous les gestionnaires de files d'attente créés sur ce serveur. Il sécurise les répertoires avec le groupe mqm local et configure un ID utilisateur pour que le service IBM MQ puisse accéder aux répertoires.

Lorsque vous sécurisez un dossier partagé avec un groupe de sécurité, un utilisateur autorisé à accéder au dossier doit disposer des données d'identification du groupe. Supposons qu'un dossier sur un serveur de fichiers distant soit sécurisé avec le groupe mqm local sur un serveur appelé *mars*. Faites de l'utilisateur qui exécute les processus de gestionnaire de files d'attente un membre du groupe mqm local sur *mars*. L'utilisateur dispose des données d'identification qui correspondent à celles du dossier sur le serveur de fichiers distant. A l'aide de ces données d'identification, le gestionnaire de files d'attente peut accéder à ses données et à ses fichiers journaux dans le dossier. L'utilisateur qui exécute des processus de gestionnaire de files d'attente sur un serveur différent est membre d'un groupe mqm local différent qui ne possède pas de données d'identification correspondantes. Lorsque le gestionnaire de files d'attente s'exécute sur un autre serveur que *mars*, il ne peut pas accéder aux données et aux fichiers journaux qu'il a créés lors de son exécution sur *mars*. Même si vous faites de l'utilisateur un utilisateur de domaine, il dispose de données d'identification différentes, car il doit obtenir les données d'identification du groupe mqm local sur *mars*, et il ne peut pas le faire à partir d'un serveur différent.

Le fait de fournir au gestionnaire de files d'attente un autre groupe de sécurité global résout le problème ; voir [Figure 73](#), à la page 580. Sécurisez un dossier distant avec un groupe global. Transmettez le nom du groupe global au gestionnaire de files d'attente lorsque vous le créez dans *mars*. Transmettez le nom de groupe global comme autre groupe de sécurité à l'aide du paramètre `-a [r]` de la commande `crtmqm`. Si vous transférez le gestionnaire de files d'attente pour qu'il s'exécute sur un autre serveur, le nom du groupe de sécurité est transféré avec lui. Le nom est transféré dans la section **AccessMode** du fichier `qm.ini` en tant que `SecurityGroup` ; par exemple :

```
AccessMode:
SecurityGroup=wmq\wmq
```

La section **AccessMode** de la `qm.ini` inclut également `RemoveMQMAccess` ; par exemple :

```
AccessMode:
RemoveMQMAccess=true/false
```

Si cet attribut est spécifié avec la valeur `true` et qu'un groupe d'accès a également été indiqué, le groupe mqm local n'a pas accès aux fichiers de données du gestionnaire de files d'attente.

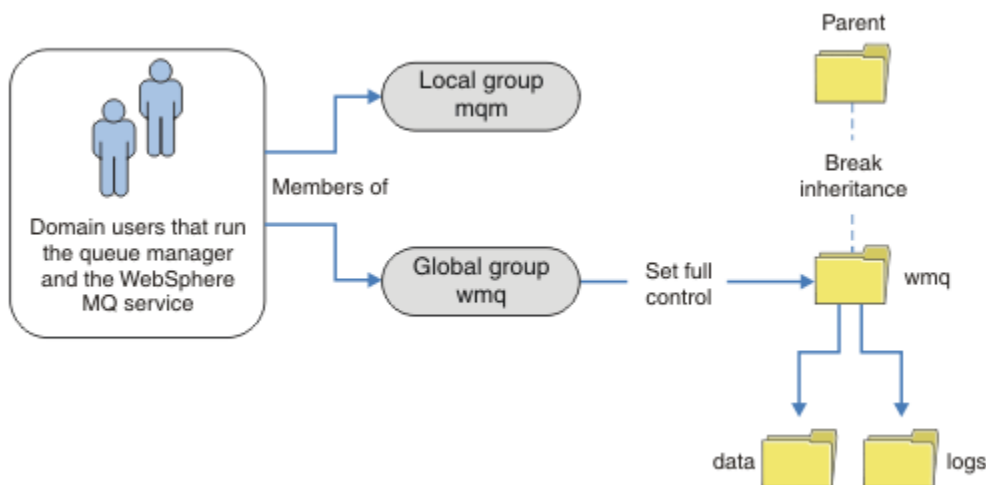


Figure 73. Sécurisation des données et des journaux du gestionnaire de files d'attente à l'aide d'un autre groupe de sécurité globale (1)

Pour que l'ID utilisateur avec lequel les processus de gestionnaire de files d'attente doivent s'exécuter dispose des données d'identification correspondantes du groupe de sécurité globale, l'ID utilisateur doit également avoir une portée globale. Vous ne pouvez pas faire d'un groupe local ou d'un principal un

membre d'un groupe global. Dans [Figure 73](#), à la page 580, les utilisateurs qui exécutent les processus de gestionnaire de files d'attente sont affichés en tant qu'utilisateurs de domaine.

Si vous déployez de nombreux serveurs IBM MQ, le regroupement d'utilisateurs dans [Figure 73](#), à la page 580 n'est pas pratique. Vous devez répéter le processus d'ajout d'utilisateurs à des groupes locaux pour chaque serveur IBM MQ. A la place, créez un groupe global `Domain mqm` sur le contrôleur de domaine et définissez les utilisateurs qui exécutent les membres IBM MQ du groupe `Domain mqm`; voir [Figure 74](#), à la page 581. Lorsque vous installez IBM MQ en tant qu'installation de domaine, `Prepare IBM MQ Wizard` fait automatiquement du groupe `Domain mqm` un membre du groupe `mqm` local. Les mêmes utilisateurs se trouvent à la fois dans les groupes globaux `Domain mqm` et `wmq`.

Conseil : Les mêmes utilisateurs peuvent exécuter IBM MQ sur des serveurs différents, mais sur un serveur individuel, vous devez avoir des utilisateurs différents pour exécuter IBM MQ en tant que service et l'exécuter de manière interactive. Vous devez également avoir des utilisateurs différents pour chaque installation sur un serveur. En règle générale, par conséquent, `Domain mqm` contient un certain nombre d'utilisateurs.

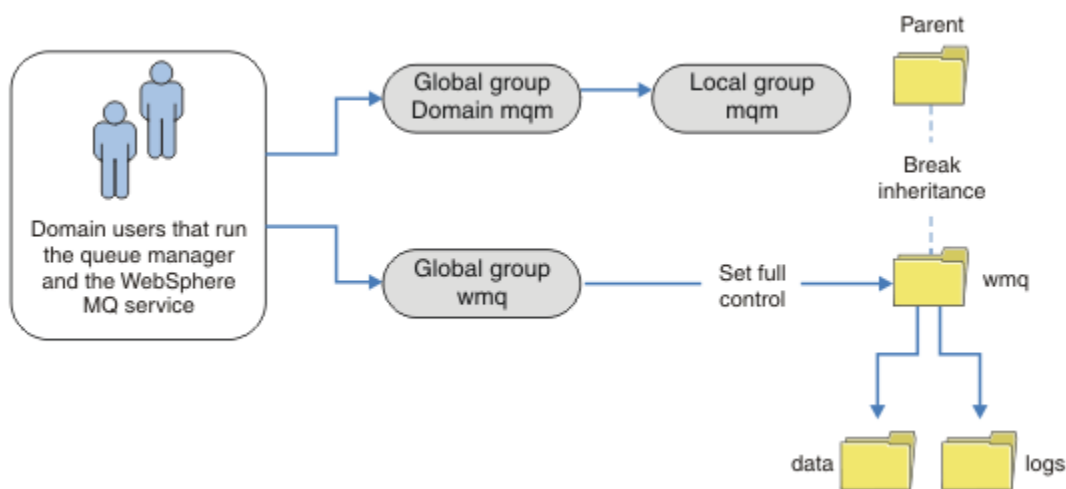


Figure 74. Sécurisation des données et des journaux du gestionnaire de files d'attente à l'aide d'un autre groupe de sécurité globale (2)

L'organisation dans [Figure 74](#), à la page 581 est inutilement compliquée en l'état. L'arrangement comporte deux groupes globaux avec des membres identiques. Vous pouvez simplifier l'organisation et définir un seul groupe global; voir [Figure 75](#), à la page 582.

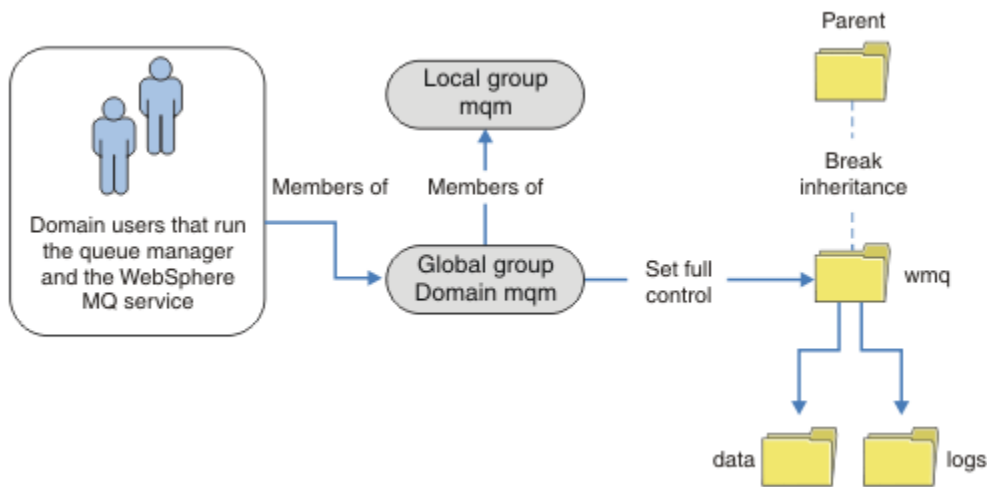


Figure 75. Sécurisation des données et des journaux du gestionnaire de files d'attente à l'aide d'un autre groupe de sécurité globale (3)

Vous pouvez également avoir besoin d'un niveau de contrôle d'accès plus fin, avec différents gestionnaires de files d'attente limités à l'accès à différents dossiers ; voir [Figure 76](#), à la page 583. Dans [Figure 76](#), à la page 583, deux groupes d'utilisateurs de domaine sont définis, dans des groupes globaux distincts pour sécuriser différents fichiers de données et journaux de gestionnaire de files d'attente. Deux groupes mqm locaux différents sont affichés, qui doivent se trouver sur des serveurs IBM MQ différents. Dans cet exemple, les gestionnaires de files d'attente sont partitionnés en deux ensembles, avec des utilisateurs différents alloués aux deux ensembles. Les deux ensembles peuvent être des gestionnaires de files d'attente de test et de production. Les autres groupes de sécurité sont appelés wmq1 et wmq2. Vous devez ajouter manuellement les groupes globaux wmq1 et wmq2 aux gestionnaires de files d'attente appropriés selon qu'ils se trouvent dans le service de test ou de production. La configuration ne peut pas tirer parti du fait que l'installation de IBM MQ propage Domain mqm au groupe mqm local comme dans [Figure 75](#), à la page 582, car il existe deux groupes d'utilisateurs.

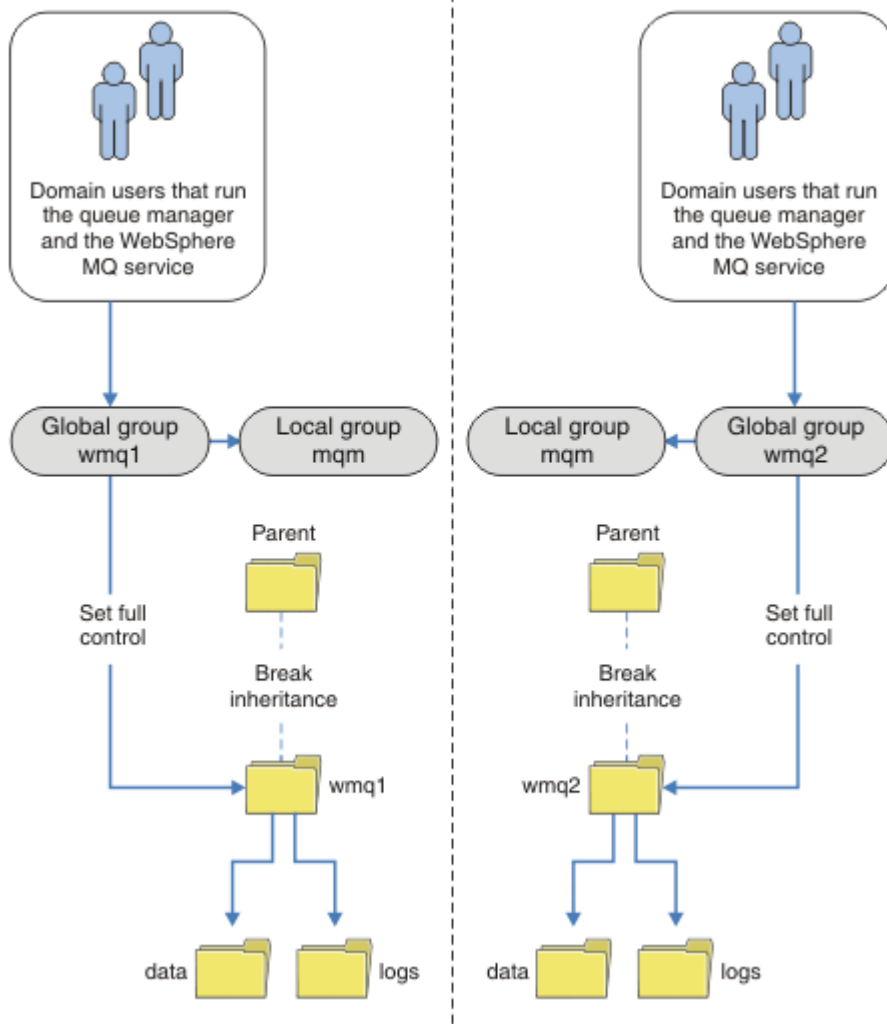


Figure 76. Sécurisation des données et des journaux du gestionnaire de files d'attente à l'aide d'un autre principal de sécurité globale (4)

Une autre façon de partitionner deux départements serait de les placer dans deux domaines de fenêtres. Dans ce cas, vous pouvez revenir à l'utilisation du modèle plus simple illustré dans la [Figure 75](#), à la page 582.

Windows *Fichiers et répertoires de données et de journaux du gestionnaire de files d'attente non partagés sécurisés sous Windows*

Cette rubrique explique comment sécuriser un autre emplacement pour les données de gestionnaire de files d'attente et les fichiers journaux, à l'aide du groupe mqm local et d'un autre groupe de sécurité.

En règle générale, vous ne configurez pas d'autre emplacement pour les données et les fichiers journaux du gestionnaire de files d'attente. Lorsque vous installez IBM MQ for Windows, le programme d'installation crée un répertoire de base de votre choix pour les gestionnaires de files d'attente créés. Il sécurise les répertoires avec le groupe mqm local et configure un ID utilisateur pour que le service IBM MQ puisse accéder aux répertoires.

Deux exemples montrent comment configurer le contrôle d'accès pour IBM MQ. Les exemples montrent comment créer un gestionnaire de files d'attente avec ses données et ses journaux dans des répertoires qui ne se trouvent pas dans les chemins de données et de journaux créés par l'installation. Dans le premier exemple, [«Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm»](#), à la page 585, vous autorisez l'accès aux répertoires de file d'attente et de journal en autorisant le groupe mqm local. Le deuxième exemple, [«Lecture et écriture de données et de fichiers journaux](#)

autorisés par un autre groupe de sécurité local», à la page 589, diffère par le fait que l'accès aux répertoires est autorisé par un autre groupe de sécurité. Lorsque les répertoires sont accessibles par un gestionnaire de files d'attente s'exécutant sur un seul serveur, la sécurisation des données et des fichiers journaux avec le groupe de sécurité alternatif vous permet de choisir de sécuriser différents gestionnaires de files d'attente avec des groupes ou des principaux locaux différents. Lorsque les répertoires sont accessibles par un gestionnaire de files d'attente s'exécutant sur des serveurs différents, par exemple avec un gestionnaire de files d'attente multi-instance, la sécurisation des données et des fichiers journaux avec le groupe de sécurité de remplacement est le seul choix ; voir «Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows», à la page 579.

La configuration des droits de sécurité des données du gestionnaire de files d'attente et des fichiers journaux n'est pas une tâche courante dans Windows. Lorsque vous installez IBM MQ for Windows, vous spécifiez des répertoires pour les données et les journaux du gestionnaire de files d'attente ou vous acceptez les répertoires par défaut. Le programme d'installation sécurise automatiquement ces répertoires avec le groupe mqm local, ce qui lui donne des droits de contrôle complets. Le processus d'installation vérifie que l'ID utilisateur qui exécute les gestionnaires de files d'attente est membre du groupe mqm local. Vous pouvez modifier les autres droits d'accès sur les répertoires pour répondre à vos besoins d'accès.

Si vous déplacez le répertoire des données et des fichiers journaux vers de nouveaux emplacements, vous devez configurer la sécurité des nouveaux emplacements. Vous pouvez modifier l'emplacement des répertoires si vous sauvegardez un gestionnaire de files d'attente et le restaurez sur un autre ordinateur, ou si vous modifiez le gestionnaire de files d'attente pour qu'il soit un gestionnaire de files d'attente multi-instance. Vous avez le choix entre deux méthodes de sécurisation des données du gestionnaire de files d'attente et des répertoires de journaux dans leur nouvel emplacement. Vous pouvez sécuriser les répertoires en limitant l'accès au groupe mqm local ou vous pouvez restreindre l'accès à n'importe quel groupe de sécurité de votre choix.

Il effectue le moins d'étapes pour sécuriser les répertoires à l'aide du groupe mqm local. Définissez les droits sur les répertoires de données et de journaux pour permettre au groupe mqm local de prendre le contrôle complet. Une approche classique consiste à copier l'ensemble de droits existant, en supprimant l'héritage du parent. Vous pouvez ensuite supprimer ou restreindre les droits d'accès d'autres principaux.

Si vous exécutez le gestionnaire de files d'attente sous un ID utilisateur différent du service configuré par l'assistant de préparation d' IBM MQ , cet ID utilisateur doit être membre du groupe mqm local. La tâche, «Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm», à la page 585, vous guide tout au long des étapes.

Vous pouvez également sécuriser les données et les fichiers journaux du gestionnaire de files d'attente à l'aide d'un autre groupe de sécurité. Le processus de sécurisation des données du gestionnaire de files d'attente et des fichiers journaux avec le groupe de sécurité alternatif comporte un certain nombre d'étapes qui font référence à Figure 77, à la page 585. Le groupe local, wmq, est un exemple de groupe de sécurité alternatif.

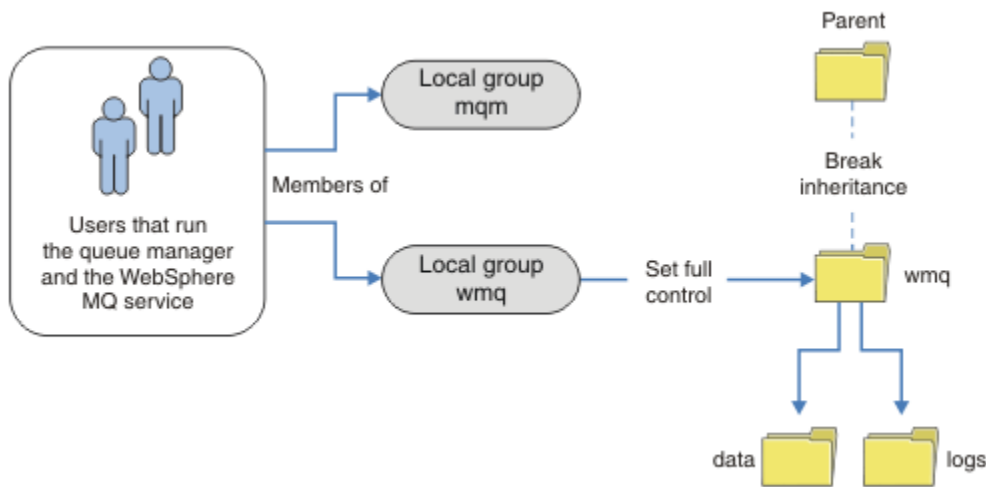


Figure 77. Sécurisation des données et des journaux du gestionnaire de files d'attente à l'aide d'un autre groupe de sécurité local, wmq

1. Créez des répertoires distincts pour les données et les journaux du gestionnaire de files d'attente, un répertoire commun ou un répertoire parent commun.
2. Copiez l'ensemble existant de droits hérités pour les répertoires ou le répertoire parent et modifiez-les en fonction de vos besoins.
3. Sécurisez les répertoires qui doivent contenir le gestionnaire de files d'attente et les journaux en accordant au groupe alternatif, wmq, des droits de contrôle complets sur les répertoires.
4. Attribuez à tous les ID utilisateur qui exécutent des processus de gestionnaire de files d'attente les données d'identification du groupe de sécurité ou du principal de remplacement:
 - a. Si vous définissez un utilisateur comme principal de sécurité de remplacement, l'utilisateur doit être le même que celui sous lequel le gestionnaire de files d'attente va s'exécuter. L'utilisateur doit être membre du groupe mqm local.
 - b. Si vous définissez un groupe local comme groupe de sécurité alternatif, ajoutez l'utilisateur sous lequel le gestionnaire de files d'attente va s'exécuter au groupe alternatif. L'utilisateur doit également être membre du groupe mqm local.
 - c. Si vous définissez un groupe global comme autre groupe de sécurité, voir «Sécurisation des données de gestionnaire de files d'attente partagées et des répertoires et fichiers journaux sous Windows», à la page 579.
5. Créez le gestionnaire de files d'attente en spécifiant l'autre groupe de sécurité ou principal dans la commande **crtmqm**, à l'aide du paramètre -a .

Windows Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm

La tâche explique comment créer un gestionnaire de files d'attente avec ses données et ses fichiers journaux stockés dans le répertoire de votre choix. L'accès aux fichiers est sécurisé par le groupe local mqm. Le répertoire n'est pas partagé.

Avant de commencer

1. Installez IBM MQ for Windows comme installation principale.
2. Exécutez Prepare IBM MQ Wizard.

Pour plus d'informations, voir [Configuration de IBM MQ avec Prepare IBM MQ Wizard](#).

Pour cette tâche, configurez l'installation pour l'exécuter avec un ID utilisateur local ou un ID utilisateur de domaine. Enfin, pour exécuter toutes les tâches dans «[Domaines et gestionnaires de](#)

files d'attente multi-instances Windows», à la page 551, l'installation doit être configurée pour un domaine.

3. Connectez-vous avec les droits d'accès administrateur pour effectuer la première partie de la tâche.

Pourquoi et quand exécuter cette tâche

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix. Elles accompagnent la tâche «[Domaines et gestionnaires de files d'attente multi-instances Windows](#)», à la page 551.

Dans Windows, vous pouvez créer la valeur par défaut de données et les chemins de journal pour un IBM MQ for Windows dans le répertoire de votre choix. L'assistant d'installation et de configuration fournit automatiquement l'accès aux répertoires au groupe local mqm et à l'ID utilisateur qui exécute les processus du gestionnaire de files d'attente. Si vous créez un gestionnaire de files d'attente en spécifiant des répertoires différents pour les fichiers de données et de journaux du gestionnaire de files d'attente, vous devez configurer des droits complets sur les répertoires.

Dans cet exemple, vous donnez au gestionnaire de files d'attente le contrôle total de ses données et de ses fichiers journaux en accordant au groupe mqm local le droit d'accès au répertoire `c:\wmq`.

La commande `crtmqm` crée un gestionnaire de files d'attente qui démarre automatiquement lorsque le poste de travail démarre avec le service IBM MQ.

Cette tâche est citée à titre d'exemple et utilise des valeurs spécifiques que vous pouvez modifier. Les valeurs que vous pouvez modifier sont en italique. A la fin de la tâche, suivez les instructions pour supprimer toutes les modifications que vous avez effectuées.

Procédure

1. Ouvrez une invite de commande.
2. Tapez la commande :

```
md c:\wmq\data, c:\wmq\logs
```

3. Définissez les droits sur les répertoires pour accorder un droit en lecture et en écriture au groupe local mqm.

```
cacls c:\wmq/T /E /G mqm:F
```

Réponse du système :

```
processed dir: c:\wmq
processed dir: c:\wmq\data
processed dir: c:\wmq\logs
```

4. Facultatif : Sélectionnez un ID qui soit membre du groupe local mqm.

Vous pouvez continuer en tant qu'administrateur, mais pour créer une configuration de production réaliste, continuez avec un ID utilisateur disposant de droits d'accès plus restreints. L'ID utilisateur doit au moins être membre du groupe mqm.

Si l'installation IBM MQ est configurée comme faisant partie d'un domaine, définissez l'ID utilisateur comme membre du groupe `Domain\mqm`. L'assistant "Préparation d'IBM MQ" définit le groupe global `Domain\mqm` comme membre du groupe local mqm, ce qui vous évite de devoir définir directement l'ID utilisateur comme membre du groupe local mqm.

5. Créez le gestionnaire de files d'attente.

```
crtmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

Réponse du système :

```
IBM MQ queue manager created.  
Directory 'c:\wmq\data\QMGR' created.  
The queue manager is associated with installation '1'  
Creating or replacing default objects for queue manager 'QMGR'  
Default objects statistics : 74 created. 0 replaced.  
Completing setup.  
Setup completed.
```

6. Vérifiez que les répertoires créés par le gestionnaire de files d'attente se trouvent dans le répertoire `c:\wmq`.

```
dir c:\wmq/D /B /S
```

7. Vérifiez que les fichiers ont les droits en lecture et en écriture, ou les droits complets sur le groupe local mqm.

```
cacls c:\wmq\*.*
```

Que faire ensuite

Testez le gestionnaire de files d'attente en insérant un message dans une file d'attente, puis en recevant un message.

1. Démarrez le gestionnaire de files d'attente.

```
strmqm QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation '1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Créez une file d'attente de test.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Réponse du système :

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)  
AMQ8006: IBM MQ queue created.  
One MQSC command read.
```

No commands have a syntax error.
All valid MQSC commands were processed.

3. Insérez un message de test à l'aide de l'exemple de programme **amqsput**.

```
echo 'A test message' | amqsput QTEST QMGR
```

Réponse du système :

```
Sample AMQSPUT0 start  
target queue is QTEST  
Sample AMQSPUT0 end
```

4. Recevez le message de test à l'aide de l'exemple de programme **amqsget**.

```
amqsget QTEST QMGR
```

Réponse du système :

```
Sample AMQSGET0 start  
message A test message  
Wait 15 seconds ...  
no more messages  
Sample AMQSGET0 end
```

5. Arrêtez le gestionnaire de files d'attente.

```
endmqm -i QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' ending.  
IBM MQ queue manager 'QMGR' ended.
```

6. Supprimez le gestionnaire de files d'attente.

```
dltmqm QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' deleted.
```

7. Supprimez les répertoires que vous avez créés.

Conseil : Ajoutez l'option /Q aux commandes pour empêcher la commande invitant à supprimer chaque fichier ou répertoire.

```
del /F /S C:\wmq\*.*  
rmdir /S C:\wmq
```

Concepts associés

«Domaines et gestionnaires de files d'attente multi-instances Windows», à la page 551

Un gestionnaire de files d'attente multi-instance sous Windows requiert le partage de ses données et de ses journaux. Le partage doit être accessible à toutes les instances du gestionnaire de files d'attente s'exécutant sur des serveurs ou des postes de travail différents. Configurez les gestionnaires de files

d'attente et partagez-les dans le cadre d'un domaine Windows . Le gestionnaire de files d'attente peut s'exécuter sur un poste de travail ou un serveur de domaine ou sur le contrôleur de domaine.

Tâches associées

Windows *Lecture et écriture de données et de fichiers journaux autorisés par un autre groupe de sécurité local*

Cette tâche explique comment utiliser l'indicateur -a dans la commande `crtmqm` . L'indicateur fournit au gestionnaire de files d'attente un autre groupe de sécurité local pour lui permettre d'accéder à ses fichiers journaux et de données.

«[Lecture et écriture de données partagées et de fichiers journaux autorisés par un autre groupe de sécurité globale](#)», à la page 566

«[Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows](#)», à la page 553

Windows *Lecture et écriture de données et de fichiers journaux autorisés par un autre groupe de sécurité local*

Cette tâche explique comment utiliser l'indicateur -a dans la commande `crtmqm` . L'indicateur fournit au gestionnaire de files d'attente un autre groupe de sécurité local pour lui permettre d'accéder à ses fichiers journaux et de données.

Avant de commencer

1. Installez IBM MQ for Windows comme installation principale.
2. Exécutez Prepare IBM MQ Wizard.

Pour plus d'informations, voir [Configuration de IBM MQ avec Prepare IBM MQ Wizard](#).

Pour cette tâche, configurez l'installation pour l'exécuter avec un ID utilisateur local ou un ID utilisateur de domaine. Enfin, pour exécuter toutes les tâches dans «[Domaines et gestionnaires de files d'attente multi-instances Windows](#)», à la page 551, l'installation doit être configurée pour un domaine.

3. Connectez-vous avec les droits d'accès administrateur pour effectuer la première partie de la tâche.

Pourquoi et quand exécuter cette tâche

Cette tâche fait partie d'un ensemble de tâches connexes qui illustrent l'accès aux données et aux fichiers journaux du gestionnaire de files d'attente. Les tâches montrent comment créer un gestionnaire de files d'attente autorisé à lire et écrire des données et des fichiers journaux stockés dans un répertoire de votre choix. Elles accompagnent la tâche «[Domaines et gestionnaires de files d'attente multi-instances Windows](#)», à la page 551.

Dans Windows, vous pouvez créer la valeur par défaut de données et les chemins de journal pour un IBM MQ for Windows dans le répertoire de votre choix. L'assistant d'installation et de configuration fournit automatiquement l'accès aux répertoires au groupe local mqm et à l'ID utilisateur qui exécute les processus du gestionnaire de files d'attente. Si vous créez un gestionnaire de files d'attente en spécifiant des répertoires différents pour les fichiers de données et de journaux du gestionnaire de files d'attente, vous devez configurer des droits complets sur les répertoires.

Dans cet exemple, vous fournissez au gestionnaire de files d'attente un autre groupe local de sécurité disposant de droits de contrôle complets sur les répertoires. Le groupe de sécurité alternatif autorise le gestionnaire de files d'attente à gérer les fichiers dans le répertoire. L'objectif principal du groupe de sécurité de remplacement est d'autoriser un groupe global de sécurité de remplacement. Utilisez un autre groupe global de sécurité pour configurer un gestionnaire de files d'attente multi-instance. Dans cet exemple, vous configurez un groupe local pour vous familiariser avec l'utilisation d'un autre groupe de sécurité sans installer IBM MQ dans un domaine. Il est inhabituel de configurer un groupe local en tant que groupe de sécurité alternatif.

Cette tâche est citée à titre d'exemple et utilise des valeurs spécifiques que vous pouvez modifier. Les valeurs que vous pouvez modifier sont en italique. A la fin de la tâche, suivez les instructions pour supprimer toutes les modifications que vous avez effectuées.

Procédure

1. Configurez un autre groupe de sécurité.

Le groupe de sécurité alternatif est généralement un groupe de domaine. Dans l'exemple, vous créez un gestionnaire de files d'attente qui utilise un autre groupe de sécurité local. Avec un autre groupe de sécurité local, vous pouvez effectuer la tâche avec une installation IBM MQ qui ne fait pas partie d'un domaine.

- a) Exécutez la commande **lusrmgr.msc** pour ouvrir la fenêtre Utilisateurs et groupes locaux.
- b) Cliquez avec le bouton droit de la souris sur **Groupes > Nouveau groupe ...**
- c) Dans la zone **Nom de groupe**, entrez *altmqm* et cliquez sur **Créer > Fermer**.
- d) Identifiez l'ID utilisateur qui exécute le service IBM MQ.
 - i) Cliquez sur **Démarrer > Exécuter ...**, tapez *services.msc* et cliquez sur **OK**.
 - ii) Cliquez sur le service IBM MQ dans la liste des services, puis cliquez sur l'onglet Connexion.
 - iii) Mémo-risez l'ID utilisateur et fermez l'explorateur de services.
- e) Ajoutez l'ID utilisateur qui exécute le service IBM MQ au groupe *altmqm*. Ajoutez également l'ID utilisateur avec lequel vous vous connectez pour créer un gestionnaire de files d'attente et l'exécuter de manière interactive.

Windows vérifie les droits du gestionnaire de files d'attente pour accéder aux répertoires de données et de journaux en vérifiant les droits de l'ID utilisateur qui exécute les processus du gestionnaire de files d'attente. L'ID utilisateur doit être un membre, directement ou indirectement via un groupe global, du groupe *altmqm* qui a autorisé les annuaires.

Si vous avez installé IBM MQ dans le cadre d'un domaine et que vous allez effectuer les tâches dans «Création d'un gestionnaire de files d'attente multi-instance sur des postes de travail de domaine ou des serveurs sous Windows», à la page 553, les ID utilisateur de domaine créés dans «Création d'un domaine Active Directory et DNS sur Windows», à la page 556 sont *wmquser1* et *wmquser2*.

Si vous n'avez pas installé le gestionnaire de files d'attente dans le cadre d'un domaine, l'ID utilisateur local par défaut qui exécute le service IBM MQ est *MUSR_MQADMIN*. Si vous avez l'intention d'effectuer les tâches sans droits d'administrateur, créez un utilisateur membre du groupe *mqm* local.

Procédez comme suit pour ajouter *wmquser1* et *wmquser2* à *altmqm*. Si votre configuration est différente, remplacez les ID utilisateur et le groupe par vos noms.

- i) Dans la liste des groupes, cliquez avec le bouton droit de la souris sur **altmqm > Propriétés > Ajouter ...**
- ii) Dans la fenêtre Sélectionner des utilisateurs, des ordinateurs ou des groupes, entrez *wmquser1 ; wmquser2* et cliquez sur **Vérifier les noms**.
- iii) Entrez le nom et le mot de passe d'un administrateur de domaine dans la fenêtre de sécurité Windows, puis cliquez sur **OK > OK > Appliquer > OK**.

2. Ouvrez une invite de commande.

3. Redémarrez le service IBM MQ.

Vous devez redémarrer le service pour que l'ID utilisateur sous lequel il s'exécute acquière les données d'identification de sécurité supplémentaires que vous avez configurées pour lui.

Entrez les commandes suivantes:

```
endmqsvc  
startmqsvc
```

Les réponses du système:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
The MQ service for installation 'Installation1' ended successfully.
```

Et:

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
The MQ service for installation 'Installation1' started successfully.
```

4. Tapez la commande :

```
md c:\wmq\data, c:\wmq\logs
```

5. Définissez les droits d'accès sur les répertoires pour autoriser l'utilisateur local *user* à accéder en lecture et en écriture.

```
cacls c:\wmq/T /E /G altmqm:F
```

Réponse du système :

```
processed dir: c:\wmq  
processed dir: c:\wmq\data  
processed dir: c:\wmq\logs
```

6. Facultatif : Sélectionnez un ID qui soit membre du groupe local *mqm*.

Vous pouvez continuer en tant qu'administrateur, mais pour créer une configuration de production réaliste, continuez avec un ID utilisateur disposant de droits d'accès plus restreints. L'ID utilisateur doit au moins être membre du groupe *mqm*.

Si l'installation IBM MQ est configurée comme faisant partie d'un domaine, définissez l'ID utilisateur comme membre du groupe Domain *mqm*. L'assistant "Préparation d'IBM MQ" définit le groupe global Domain *mqm* comme membre du groupe local *mqm*, ce qui vous évite de devoir définir directement l'ID utilisateur comme membre du groupe local *mqm*.

7. Créez le gestionnaire de files d'attente.

```
crtmqm -a altmqm -sax -u SYSTEM.DEAD.LETTER.QUEUE -md c:\wmq\data -ld c:\wmq\logs QMGR
```

Réponse du système :

```
IBM MQ queue manager created.  
Directory 'c:\wmq1\data\QMGR' created.  
The queue manager is associated with installation '1'  
Creating or replacing default objects for queue manager 'QMGR'  
Default objects statistics : 74 created. 0 replaced.  
Completing setup.  
Setup completed.
```

8. Vérifiez que les répertoires créés par le gestionnaire de files d'attente se trouvent dans le répertoire *c:\wmq*.

```
dir c:\wmq/D /B /S
```

9. Vérifiez que les fichiers ont les droits en lecture et en écriture, ou les droits complets sur le groupe local *mqm*.

```
cacls c:\wmq\*.*
```

Que faire ensuite

Testez le gestionnaire de files d'attente en insérant un message dans une file d'attente, puis en recevant un message.

1. Démarrez le gestionnaire de files d'attente.

```
strmqm QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' starting.  
The queue manager is associated with installation '1'.  
5 log records accessed on queue manager 'QMGR' during the log  
replay phase.  
Log replay for queue manager 'QMGR' complete.  
Transaction manager state recovered for queue manager 'QMGR'.  
IBM MQ queue manager 'QMGR' started using V7.1.0.0.
```

2. Créez une file d'attente de test.

```
echo define qlocal(QTEST) | runmqsc QMGR
```

Réponse du système :

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QMGR.
```

```
1 : define qlocal(QTEST)  
AMQ8006: IBM MQ queue created.  
One MQSC command read.  
No commands have a syntax error.  
All valid MQSC commands were processed.
```

3. Insérez un message de test à l'aide de l'exemple de programme **amqsput**.

```
echo 'A test message' | amqsput QTEST QMGR
```

Réponse du système :

```
Sample AMQSPUT0 start  
target queue is QTEST  
Sample AMQSPUT0 end
```

4. Recevez le message de test à l'aide de l'exemple de programme **amqsget**.

```
amqsget QTEST QMGR
```

Réponse du système :

```
Sample AMQSGET0 start
```



```
message A test message
Wait 15 seconds ...
no more messages
Sample AMQSGETO end
```

5. Arrêtez le gestionnaire de files d'attente.

```
endmqm -i QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' ending.
IBM MQ queue manager 'QMGR' ended.
```

6. Supprimez le gestionnaire de files d'attente.

```
dltmqm QMGR
```

Réponse du système :

```
IBM MQ queue manager 'QMGR' deleted.
```

7. Supprimez les répertoires que vous avez créés.

Conseil : Ajoutez l'option /Q aux commandes pour empêcher la commande invitant à supprimer chaque fichier ou répertoire.

```
del /F /S C:\wmq\*. *
rmdir /S C:\wmq
```

Tâches associées

Windows

Lecture et écriture de données et de fichiers journaux autorisées par le groupe local mqm. La tâche explique comment créer un gestionnaire de files d'attente avec ses données et ses fichiers journaux stockés dans le répertoire de votre choix. L'accès aux fichiers est sécurisé par le groupe local mqm. Le répertoire n'est pas partagé.

Linux

Création d'un gestionnaire de files d'attente multi-instance sous Linux

Exemple illustrant la configuration d'un gestionnaire de files d'attente multi-instance sous Linux. La configuration est petite pour illustrer les concepts impliqués. L'exemple est basé sur Linux Red Hat Enterprise 5. Les étapes diffèrent sur les autres plateformes UNIX .

Pourquoi et quand exécuter cette tâche

L'exemple est configuré sur un ordinateur portable 2 GHz avec 3 Go de mémoire RAM exécutant Windows 7 Service Pack 1. Deux machines virtuelles VMware , Server1 et Server2, exécutent des images Linux Red Hat Enterprise 5 en 640 Mo. Server1 héberge le système de fichiers réseau (NFS), les journaux du gestionnaire de files d'attente et une instance à haute disponibilité. Il n'est pas habituel que le serveur NFS héberge également l'une des instances de gestionnaire de files d'attente ; il s'agit de simplifier l'exemple. Server2 monte les journaux du gestionnaire de files d'attente Server1 avec une instance de secours. Un client WebSphere MQ MQI est installé sur une image VMware 400 Mo supplémentaire qui exécute Windows 7 Service Pack 1 et exécute les exemples d'applications à haute disponibilité. Toutes les machines virtuelles sont configurées dans le cadre d'un réseau hôte VMware uniquement pour des raisons de sécurité.

Remarque : Vous devez placer uniquement les données du gestionnaire de files d'attente sur un serveur NFS . Sur le système NFS, utilisez les trois options suivantes avec la commande de montage pour sécuriser le système:

- **noexec**
En utilisant cette option, vous arrêtez l'exécution des fichiers binaires sur le système NFS, ce qui empêche un utilisateur distant d'exécuter du code indésirable sur le système.
- **nosuid**
En utilisant cette option, vous empêchez l'utilisation des bits set-user-identifier et set-group-identifier, ce qui empêche un utilisateur distant d'obtenir des privilèges plus élevés.
- **nodev**
En utilisant cette option, vous arrêtez l'utilisation ou la définition de caractères et bloquez les unités spéciales, ce qui empêche un utilisateur distant de sortir d'une prison chroot.

Procédure

1. Connectez-vous en tant qu'utilisateur root.
2. Lisez la rubrique [Installation de IBM MQ -présentation](#) et suivez le lien approprié pour installer IBM MQ, créer l'utilisateur et le groupe mqm et définir /var/mqm.
3. Effectuez la tâche [Vérification du comportement du système de fichiers partagé](#) pour vérifier que le système de fichiers prend en charge les [gestionnaires de files d'attente multi-instance](#).
4. Pour Server1, procédez comme suit:
 - a. Créez des répertoires de journaux et de données dans un dossier commun, /MQHA, qui doit être partagé. Exemple :
 - i) **mkdir** /MQHA
 - ii) **mkdir** /MQHA/logs
 - iii) **mkdir** /MQHA/qmgrs
5. Pour Server2, procédez comme suit:
 - a. Créez le dossier /MQHA pour monter le système de fichiers partagé. Conservez le même chemin que sur Server1. Exemple :
 - i) **mkdir** /MQHA
6. Vérifiez que les répertoires MQHA appartiennent à l'utilisateur et au groupe mqm et que les droits d'accès sont définis sur rwx pour l'utilisateur et le groupe. Par exemple, **ls -al** affiche d_{rwx}rwxr-x_r mqm mqm 4096 Nov 27 14:38 MQDATA .
 - a. **chown -R** mqm:mqm /MQHA
 - b. **chmod -R** ug+rwx /MQHA
7. Créez le gestionnaire de files d'attente en entrant la commande suivante: **crtmqm -ld /MQHA/logs -md /MQHA/qmgrs QM1**
8. Ajouter²/MQHA *(rw, sync, no_wdelay, fsid=0) à /etc/exports
9. Pour Server1, procédez comme suit:
 - a. Démarrez le démon NFS : **/etc/init.d/ nfs start**
 - b. Copiez les détails de configuration du gestionnaire de files d'attente à partir de Server1:

```
dspmqlinf -o command QM1
```

et copiez le résultat dans le presse-papiers:

```
addmqinf -s QueueManager
-v Name=QM1
-v Directory=QM1
-v Prefix=/var/mqm
-v DataPath=/MQHA/qmgrs/QM1
```

² Le '*' permet à toutes les machines pouvant accéder à ce montage / MQHA d'accéder en lecture / écriture. Limitez l'accès à une machine de production.

10. Pour Server2, procédez comme suit:

- a. Montez le système de fichiers exporté /MQHA en entrant la commande suivante: **mount -t nfs4 -o hard,intr Server1:/ /MQHA**
- b. Collez la commande de configuration du gestionnaire de files d'attente dans Server2:

```
addmqinf -s QueueManager
-v Name=QM1
-v Directory=QM1
-v Prefix=/var/mqm
-v DataPath=/MQHA/qmgrs/QM1
```

11. Démarrez les instances de gestionnaire de files d'attente, dans l'un ou l'autre ordre, à l'aide du paramètre-x : **strmqm -x QM1**.

La commande utilisée pour démarrer les instances de gestionnaire de files d'attente doit être émise à partir de la même installation IBM MQ que la commande **addmqinf**. Pour démarrer et arrêter le gestionnaire de files d'attente à partir d'une autre installation, vous devez d'abord définir l'installation associée au gestionnaire de files d'attente à l'aide de la commande **setmqm**. Pour plus d'informations, voir [setmqm](#).

Linux Vérification du gestionnaire de files d'attente multi-instance sous Linux

Utilisez les exemples de programme **amqsgbac**, **amqspbac** et **amqsmbac** pour vérifier une configuration de gestionnaire de files d'attente multi-instance. Cette rubrique fournit un exemple de configuration permettant de vérifier une configuration de gestionnaire de files d'attente multi-instance sur Linux Red Hat Enterprise 5.

Les exemples de programmes à haute disponibilité utilisent la reconnexion automatique du client. Lorsque le gestionnaire de files d'attente connecté échoue, le client tente de se reconnecter à un gestionnaire de files d'attente du même groupe de gestionnaires de files d'attente. La description des exemples, [Exemples de programmes à haute disponibilité](#), illustre la reconnexion du client à l'aide d'un gestionnaire de files d'attente à instance unique pour plus de simplicité. Vous pouvez utiliser les mêmes exemples avec des gestionnaires de files d'attente multi-instance pour vérifier une configuration de gestionnaire de files d'attente multi-instance.

L'exemple utilise la configuration multi-instance décrite dans «[Création d'un gestionnaire de files d'attente multi-instance sous Linux](#)», à la page 593. Utilisez la configuration pour vérifier que le gestionnaire de files d'attente multi-instance bascule sur l'instance de secours. Arrêtez le gestionnaire de files d'attente à l'aide de la commande **endmqm** et utilisez l'option -s, switchover,. Les programmes client se reconnectent à la nouvelle instance de gestionnaire de files d'attente et continuent à utiliser la nouvelle instance après un léger délai.

Dans l'exemple, le client s'exécute sur un système Windows 7 Service Pack 1. Le système héberge deux serveurs VMware Linux qui exécutent le gestionnaire de files d'attente multi-instance.

Vérification de la reprise en ligne avec IBM MQ Explorer

Avant d'utiliser les modèles d'application pour vérifier la reprise en ligne, exécutez le IBM MQ Explorer sur chaque serveur. Ajoutez les deux instances de gestionnaire de files d'attente à chaque explorateur à l'aide de l'assistant **Ajout d'un gestionnaire de files d'attente distantes > Connexion directe à un gestionnaire de files d'attente multi-instance**. Vérifiez que les deux instances sont en cours d'exécution, en autorisant la mise en veille. Fermez la fenêtre exécutant l'image VMware avec l'instance active, mettez virtuellement le serveur hors tension ou arrêtez l'instance active, ce qui permet de basculer vers l'instance de secours.

Remarque : Si vous mettez le serveur hors tension, assurez-vous qu'il n'est pas celui qui héberge /MQHA !

Remarque : L'option **Autoriser le basculement vers une instance de secours** peut ne pas être disponible dans la boîte de dialogue **Arrêter un gestionnaire de files d'attente**. L'option est manquante car le gestionnaire de files d'attente s'exécute en tant que gestionnaire de files d'attente à instance unique. Vous devez l'avoir démarré sans l'option **Autoriser une instance de secours**. Si votre demande d'arrêt

du gestionnaire de files d'attente est rejetée, consultez la fenêtre **Détails** car il est possible qu'aucune instance de secours ne soit en cours d'exécution.

Vérification de la reprise en ligne à l'aide des exemples de programme

Choisissez un serveur à utiliser pour exécuter l'instance active

Vous avez peut-être choisi l'un des serveurs pour héberger le répertoire MQHA ou le système de fichiers. Si vous prévoyez de tester la reprise en ligne en fermant la fenêtre VMware qui exécute le serveur actif, assurez-vous qu'il ne s'agit pas de celui qui héberge MQHA !

Sur le serveur exécutant l'instance de gestionnaire de files d'attente active

Remarque : L'exécution du canal SVRCONN avec le paramètre MCAUSER défini sur mqm permet de réduire le nombre d'étapes de configuration dans l'exemple. Si un autre ID utilisateur est choisi et que votre système est configuré différemment de celui utilisé dans l'exemple, vous risquez de rencontrer des problèmes de droits d'accès. N'utilisez pas mqm en tant que MCAUSER sur un système exposé ; cela risque de compromettre considérablement la sécurité.

1. Modifiez *ipaddr1* et *ipaddr2* et sauvegardez les commandes suivantes dans */MQHA/hasamples.tst*.

```
DEFINE QLOCAL(SOURCE) REPLACE
DEFINE QLOCAL(TARGET) REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(SVRCONN) TRPTYPE(TCP) +
MCAUSER('mqm') REPLACE
DEFINE CHANNEL(CHANNEL1) CHLTYPE(CLNTCONN) TRPTYPE(TCP) +
CONNAME(' ipaddr1 (1414), ipaddr2
(1414)') QMNAME(QM1) REPLACE
START CHANNEL(CHANNEL1)
DEFINE LISTENER(LISTENER.TCP) TRPTYPE(TCP) CONTROL(QMGR)
DISPLAY LISTENER(LISTENER.TCP) CONTROL
START LISTENER(LISTENER.TCP)
DISPLAY LSSTATUS(LISTENER.TCP) STATUS
```

2. Ouvrez une fenêtre de terminal avec le chemin */MQHA* et exécutez la commande suivante:

```
runmqsc -m QM1 < hasamples.tst
```

3. Vérifiez que le programme d'écoute est en cours d'exécution et qu'il a le contrôle du gestionnaire de files d'attente, en examinant la sortie de la commande **runmqsc** .

```
LISTENER(LISTENER.TCP)CONTROL(QMGR)
LISTENER(LISTENER.TCP)STATUS(RUNNING)
```

Ou bien, en utilisant le IBM MQ Explorer que le programme d'écoute TCPIP est en cours d'exécution et possède `Control = Queue Manager`.

Sur le client

1. Copiez la table de connexion client AMQCLCHL.TAB de */MQHA/qmgrs/QM1.000/@ipcc* sur le serveur vers *C:* sur le client.
2. Ouvrez une invite de commande avec le chemin *C:* et définissez la variable d'environnement *MQCHLLIB* pour qu'elle pointe vers la table de définition de canal du client (CCDT)

```
SET MQCHLLIB=C:\
```

3. A l'invite de commande, entrez les commandes suivantes:

```
start amqsghac TARGET QM1
start amqsmhac -s SOURCE -t TARGET -m QM1
start amqsp hac SOURCE QM1
```

Sur le serveur exécutant l'instance de gestionnaire de files d'attente active

1. L'un ou l'autre :
 - Fermez la fenêtre exécutant l'image VMware avec l'instance de serveur active.
 - A l'aide de la IBM MQ Explorer, arrêtez l'instance de gestionnaire de files d'attente active, en autorisant la commutation vers l'instance de secours et en demandant aux clients reconnectables de se reconnecter.
2. Les trois clients détectent finalement que la connexion est interrompue, puis se reconnectent. Dans cette configuration, si vous fermez la fenêtre du serveur, le rétablissement des trois connexions prend environ sept minutes. Certaines connexions sont rétablies bien avant d'autres.

Résultats

```
N:\>amqsphac SOURCE QM1
Sample AMQSPHAC start
target queue is SOURCE
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

```
N:\>amqsmhac -s SOURCE -t TARGET -m QM1
Sample AMQSMHA0 start

17:05:25 : EVENT : Connection Reconnecting (Delay: 97ms)
17:05:48 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:53 : EVENT : Connection Reconnected
```

```
N:\>amqsgnac TARGET QM1
Sample AMQSGHAC start
message Message 1
message Message 2
message Message 3
message Message 4
message Message 5
17:05:25 : EVENT : Connection Reconnecting (Delay: 156ms)
17:05:47 : EVENT : Connection Reconnecting (Delay: 0ms)
17:05:52 : EVENT : Connection Reconnected
message Message 6
message Message 7
message Message 8
message Message 9
```

Conversion d'une instance unique en gestionnaire de files d'attente multi-instance sous AIX and Linux

Pour convertir un gestionnaire de files d'attente à instance unique en gestionnaire de files d'attente multi-instance sous AIX and Linux, vous devez déplacer les données du gestionnaire de files d'attente dans un répertoire partagé et reconfigurer le gestionnaire de files d'attente sur deux autres serveurs.

Avant de commencer

Vous devez vérifier les prérequis pour l'exécution d'un gestionnaire de files d'attente multi-instance dans le cadre de cette tâche. Pour obtenir la liste des environnements testés, voir [Instruction de test pour les systèmes de fichiers du gestionnaire de files d'attente multi-instance IBM MQ](#). Les autres environnements peuvent fonctionner ; un outil de test est fourni avec IBM MQ pour vous aider à qualifier d'autres environnements.

Vous devez disposer de trois serveurs pour exécuter un gestionnaire de files d'attente multi-instance. Un serveur dispose d'un système de fichiers partagé pour stocker les données et journaux du gestionnaire de files d'attente. Les autres serveurs exécutent les instances actives et de secours du gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Vous disposez d'un gestionnaire de files d'attente mono-instance que vous souhaitez convertir en un gestionnaire de files d'attente multi-instance. La conversion du gestionnaire de files d'attente est simple mais vous devez exécuter d'autres tâches pour créer un environnement de production entièrement automatisé.

Vous devez vérifier les conditions prérequis d'un gestionnaire de files d'attente multi-instance, configurer l'environnement et le vérifier. Vous devez également configurer un système de surveillance et de gestion pour détecter si le gestionnaire de files d'attente multi-instance a échoué et a été redémarré automatiquement. Identifiez ensuite l'incident à l'origine du redémarrage, corrigez-le et redémarrez l'instance de secours. Vous devez également modifier les applications ou le mode de connexion des applications au gestionnaire de files d'attente, de sorte qu'elles puissent reprendre le traitement après le redémarrage du gestionnaire de files d'attente.

Procédure

1. Vérifiez le système d'exploitation sur lequel vous allez exécuter le gestionnaire de files d'attente et le système de fichiers sur lequel les données et journaux du gestionnaire de files d'attente sont stockés. Assurez-vous qu'ils peuvent exécuter un gestionnaire de files d'attente multi-instance.
 - a) Voir [Testing statement for IBM MQ multi-instance queue manager file systems](#). Vérifiez si la combinaison de système d'exploitation et de système de fichiers a été testée et si elle est en mesure d'exécuter un gestionnaire de files d'attente multi-instance.

Un système de fichiers partagé doit fournir un verrouillage basé sur bail pour pouvoir exécuter des gestionnaires de files d'attente multi-instances. Ce type de verrouillage est une fonction récente de certains systèmes de fichiers partagés et des correctifs sont requis dans certains cas. La déclaration de prise en charge vous fournit les informations essentielles.
 - b) Exécutez **amqmfsc** pour vérifier que le système de fichiers est correctement configuré.

La configuration des systèmes de fichiers privilégie parfois les performances par rapport à l'intégrité des données. Il est important de vérifier la configuration du système de fichiers. Un rapport négatif de l'outil **amqmfsc** vous informe que les paramètres ne sont pas valides. Un résultat positif indique que le système de fichiers est approprié ; ce résultat n'est cependant pas définitif. Il s'agit d'une indication fiable.
 - c) Exécutez l'application de vérification d'intégrité fournie dans la note technique [Testing a shared file system for compatibility with IBM MQ Multi-instance Queue Managers](#).

Cette application vérifie si le gestionnaire de files d'attente redémarre correctement.
2. Configurez un utilisateur et un groupe pour pouvoir accéder à une ressource partagée sur le système de fichiers en réseau à partir de chaque serveur qui exécute une instance de gestionnaire de files d'attente.

Sous AIX and Linux, `uid` et `gid` for `mqm` dans `/etc/passwd` doivent être identiques sur chaque système ; voir [Création d'un gestionnaire de files d'attente multi-instance sur Linux](#) .
3. Configurez un répertoire pour la ressource partagée sur le système de fichiers en réseau avec les droits d'accès appropriés.

Une configuration standard permet de configurer un seul répertoire partagé qui contient tous les répertoires de données et de journaux pour tous les gestionnaires de files d'attente qui utilisent le disque partagé ; voir [Partage de répertoires qmgrs et de journaux nommés dans Exemple directory configurations on AIX and Linux systems](#).

Par exemple, créez un répertoire racine sur la ressource partagée appelée MQHA qui comporte des sous-répertoires `data` et `logs`. Chaque gestionnaire de files d'attente crée ses propres répertoires de données et de journaux sous `data` et `logs`. Créez `/MQHA` sur l'unité partagée. `/MQHA` appartient à l'utilisateur et au groupe `mqm` et dispose des droits d'accès `rwx`.

4. Copiez les données et journaux du gestionnaire de files d'attente dans la ressource partagée.

Suivez la procédure de sauvegarde du gestionnaire de files d'attente décrite dans [Sauvegarde des données de gestionnaire de files d'attente](#).

Remarque : L'utilitaire `hamvmqm` peut être utilisé sous Windows, mais pas sous AIX and Linux.

5. Mettez à jour les informations de configuration du gestionnaire de files d'attente stockées sur le serveur du gestionnaire de files d'attente en cours comme suit :

- a) Modifiez la strophe `Log` : dans le fichier `qm.ini` du gestionnaire de files d'attente, qui se trouve sur `share` :

```
LogPath= share/logs/QMgrName
```

- b) Modifiez la strophe `QueueManager` : dans le fichier `IBM MQ mqs.ini`, qui se trouve généralement dans le répertoire `/var/mqm` sous AIX and Linux :

```
DataPath= share/data/QMgrName
```

Où `QMgrName` est le nom `Directory` dans la strophe `QueueManager` : du fichier `mqs.ini` et `share` est l'espace de partage dans lequel les données et les journaux sont déplacés.

6. Ajoutez les informations de configuration du gestionnaire de files d'attente au nouveau serveur du gestionnaire de files d'attente.

- a) Exécutez la commande **dspmqlinf** pour afficher les informations de gestionnaire de files d'attente. Exécutez la commande sur le serveur qui a exécuté le gestionnaire de files d'attente.

```
dspmqlinf -o command QMgrName
```

Le résultat de la commande permet de créer la configuration d'un gestionnaire de files d'attente.

```
addmqinf -s QueueManager -v Name= QMgrName -v Directory= QMgrName -v  
Prefix=d:\var\mqm Datapath= \share\data\QMgrName
```

- b) Créez la configuration d'un gestionnaire de files d'attente sur l'autre serveur. Exécutez la commande **addmqinf** copiée depuis la sortie précédente.

7. Ajoutez l'adresse réseau du nouveau serveur au nom de connexion dans les définitions de client et de canal.

- a) Recherchez tous les paramètres TCPIP client, émetteur et demandeur qui se rapportent au serveur.

Il se peut que les paramètres client se trouvent dans la table de définition de canal du client (CCDT), dans les variables d'environnement, dans les fichiers de propriétés Java ou dans le code client. Les canaux de cluster détectent automatiquement le nom de connexion du gestionnaire de files d'attente dans son canal récepteur de cluster. Si le nom de ce canal est vide ou omis, TCPIP recherche l'adresse IP du serveur qui héberge le gestionnaire de files d'attente.

- b) Modifiez le nom de connexion de chaque connexion pour y inclure les adresses TCPIP des deux serveurs qui hébergent le gestionnaire de files d'attente multi-instance. Par exemple, remplacez le nom de connexion suivant :

```
echo DISPLAY CHANNEL(ENGLAND) CONNAME | runmqsc QM1
```

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QM1.  
1: DISPLAY CHANNEL(ENGLAND) CONNAME  
AMQ8414: Display Channel details.  
CHANNEL(ENGLAND) CHLTYPE(SDR)  
CONNAME(LONDON)
```

par :

```
echo ALTER CHANNEL(ENGLAND) CHLTYPE(SDR) CONNAME('LONDON, BRISTOL') | runmqsc QM1
```

8. Mettez à jour vos procédures de surveillance et de gestion pour détecter le redémarrage du gestionnaire de files d'attente.
9. Mettez à jour les applications client pour qu'elles se reconnectent automatiquement, le cas échéant.
10. Mettez à jour la procédure de démarrage de vos applications IBM MQ pour qu'elles démarrent en tant que services de gestionnaire de files d'attente.
11. Démarrez chaque instance du gestionnaire de files d'attente, et attribuez-leur une haute disponibilité. La première instance du gestionnaire de files d'attente qui est démarrée devient l'instance active. Exécutez la commande deux fois, une fois sur chaque serveur.

```
strmqm -x QMgrName
```

Que faire ensuite

Pour que les gestionnaires de files d'attente multi-instances disposent de la plus haute disponibilité, vous devez créer des applications client qui peuvent se reconnecter et des applications serveur qui peuvent être redémarrées ; voir [Récupération d'application](#).

Concepts associés

[Récupération d'application](#)

[reconnexion client automatique](#)

[Reconnexion canal et client](#)

[Gestionnaires de files d'attente multi-instance](#)

[Fichiers de configuration du gestionnaire de files d'attente, qm.ini](#)

[Système de fichiers partagé](#)

Tâches associées

[Sauvegarde des données de gestionnaire de files d'attente](#)

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

[Création d'un gestionnaire de files d'attente multi-instance sous Linux](#)

[Transfert d'un gestionnaire de files d'attente vers le stockage MSCS](#)

[Vérification du verrouillage du système de fichiers partagé](#)

Référence associée


[amqmfsc \(vérification du système de fichiers\)](#)

[Fichier de configuration IBM MQ, mqs.ini](#)

Information associée

[Test d'un système de fichiers partagé à des fins de compatibilité avec les gestionnaires de files d'attente multi-instances IBM MQ](#)

[Instruction de test pour les systèmes de fichiers de gestionnaire de files d'attente multi-instance IBM MQ](#)

 *Conversion d'une instance unique en gestionnaire de files d'attente multi-instance sous Windows*

Pour convertir un gestionnaire de files d'attente à instance unique en gestionnaire de files d'attente multi-instance, sur les plateformes Windows, vous devez déplacer les données du gestionnaire de files

d'attente dans un répertoire partagé et reconfigurer le gestionnaire de files d'attente sur deux autres serveurs.

Avant de commencer

Vous devez vérifier les conditions prérequis de l'exécution d'un gestionnaire de files d'attente multi-instance dans le cadre de cette tâche. Pour obtenir la liste des environnements testés, voir [Instruction de test pour les systèmes de fichiers du gestionnaire de files d'attente multi-instance IBM MQ](#). Les autres environnements peuvent fonctionner ; un outil de test est fourni avec IBM MQ pour vous aider à qualifier d'autres environnements.

Vous devez disposer de trois serveurs pour exécuter un gestionnaire de files d'attente multi-instance. Un serveur dispose d'un système de fichiers partagé pour stocker les données et journaux du gestionnaire de files d'attente. Les autres serveurs exécutent les instances actives et de secours du gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Vous disposez d'un gestionnaire de files d'attente mono-instance que vous souhaitez convertir en un gestionnaire de files d'attente multi-instance. La conversion du gestionnaire de files d'attente est simple mais vous devez exécuter d'autres tâches pour créer un environnement de production entièrement automatisé.

Vous devez vérifier les conditions prérequis d'un gestionnaire de files d'attente multi-instance, configurer l'environnement et le vérifier. Vous devez également configurer un système de surveillance et de gestion pour détecter si le gestionnaire de files d'attente multi-instance a échoué et a été redémarré automatiquement. Identifiez ensuite l'incident à l'origine du redémarrage, corrigez-le et redémarrez l'instance de secours. Vous devez également modifier les applications ou le mode de connexion des applications au gestionnaire de files d'attente, de sorte qu'elles puissent reprendre le traitement après le redémarrage du gestionnaire de files d'attente.

Procédure

1. Vérifiez le système d'exploitation sur lequel vous allez exécuter le gestionnaire de files d'attente et le système de fichiers sur lequel les données et journaux du gestionnaire de files d'attente sont stockés. Assurez-vous qu'ils peuvent exécuter un gestionnaire de files d'attente multi-instance.
 - a) Voir [Testing statement for IBM MQ multi-instance queue manager file systems](#). Vérifiez si la combinaison de système d'exploitation et de système de fichiers a été testée et si elle est en mesure d'exécuter un gestionnaire de files d'attente multi-instance.

Un système de fichiers partagé doit fournir un verrouillage basé sur bail pour pouvoir exécuter des gestionnaires de files d'attente multi-instances. Ce type de verrouillage est une fonction récente de certains systèmes de fichiers partagés et des correctifs sont requis dans certains cas. La déclaration de prise en charge vous fournit les informations essentielles.
 - b) Exécutez l'application de vérification d'intégrité fournie dans la note technique [Testing a shared file system for compatibility with IBM MQ Multi-instance Queue Managers](#).

Cette application vérifie si le gestionnaire de files d'attente redémarre correctement.
2. Configurez un utilisateur et un groupe pour pouvoir accéder à une ressource partagée sur le système de fichiers en réseau à partir de chaque serveur qui exécute une instance de gestionnaire de files d'attente.

Sous Windows, les ID de sécurité du groupe mqm peuvent être différents, voir Domaines [Windows et gestionnaires de files d'attente multi-instance](#).
3. Configurez un répertoire pour la ressource partagée sur le système de fichiers en réseau avec les droits d'accès appropriés.

Une configuration standard consiste à définir un répertoire partagé unique qui contient tous les répertoires de données et de journaux pour tous les gestionnaires de files d'attente qui utilisent le disque partagé ; voir [Partage de répertoires qmgrs et de journaux nommés](#).

Par exemple, créez un répertoire racine sur la ressource partagée appelée MQHA qui comporte des sous-répertoires data et logs. Chaque gestionnaire de files d'attente crée ses propres répertoires de données et de journaux sous data et logs. Créez *drive* \MQHA sur l'unité partagée. Le propriétaire est membre de mqm. mqm doit disposer des droits de contrôle complets. Créez une ressource partagée pour *drive*\MQHA.

4. Copiez les données et journaux du gestionnaire de files d'attente dans la ressource partagée. Sous Windows, vous pouvez exécuter la commande **hamvmqm** pour déplacer les données du gestionnaire de files d'attente vers la ressource partagée.
5. Ajoutez les informations de configuration du gestionnaire de files d'attente au nouveau serveur du gestionnaire de files d'attente.
 - a) Exécutez la commande **dspmqlnf** pour afficher les informations relatives au gestionnaire de files d'attente
Exécutez la commande sur le serveur qui a exécuté le gestionnaire de files d'attente.

```
dspmqlnf -o command QMgrName
```

Le résultat de la commande permet de créer la configuration d'un gestionnaire de files d'attente.

```
addmqinf -s QueueManager -v Name= QMgrName -v Directory= QMgrName -v  
Prefix=d:\var\mqm Datapath= \share\data\QMgrName
```

- b) Créez la configuration d'un gestionnaire de files d'attente sur l'autre serveur.
Exécutez la commande **addmqinf** copiée depuis la sortie précédente.
6. Ajoutez l'adresse réseau du nouveau serveur au nom de connexion dans les définitions de client et de canal.
 - a) Recherchez tous les paramètres TCPIP client, émetteur et demandeur qui se rapportent au serveur.
 - Il se peut que les paramètres client se trouvent dans la table de définition de canal du client (CCDT), dans les variables d'environnement, dans les fichiers de propriétés Java ou dans le code client.
 - Les canaux de cluster détectent automatiquement le nom de connexion du gestionnaire de files d'attente dans son canal récepteur de cluster. Si le nom de ce canal est vide ou omis, TCPIP recherche l'adresse IP du serveur qui héberge le gestionnaire de files d'attente.
 - b) Modifiez le nom de connexion de chaque connexion pour y inclure les adresses TCPIP des deux serveurs qui hébergent le gestionnaire de files d'attente multi-instance.
Par exemple, remplacez le nom de connexion suivant :

```
echo DISPLAY CHANNEL(ENGLAND) CONNAME | runmqsc QM1
```

```
5724-H72 (C) Copyright IBM Corp. 1994, 2024. ALL RIGHTS RESERVED.  
Starting MQSC for queue manager QM1.  
1: DISPLAY CHANNEL(ENGLAND) CONNAME  
AMQ8414: Display Channel details.  
CHANNEL(ENGLAND) CHLTYPE(SDR)  
CONNAME(LONDON)
```

par :

```
echo ALTER CHANNEL(ENGLAND) CHLTYPE(SDR) CONNAME('LONDON, BRISTOL') | runmqsc QM1
```

7. Mettez à jour vos procédures de surveillance et de gestion pour détecter le redémarrage du gestionnaire de files d'attente.
8. Mettez à jour les applications client pour qu'elles se reconnectent automatiquement, le cas échéant.
9. Mettez à jour la procédure de démarrage de vos applications IBM MQ pour qu'elles démarrent en tant que services de gestionnaire de files d'attente.

10. Démarrez chaque instance du gestionnaire de files d'attente, et attribuez-leur une haute disponibilité. La première instance du gestionnaire de files d'attente qui est démarrée devient l'instance active. Exécutez la commande deux fois, une fois sur chaque serveur.

```
startmqm -x QMgrName
```

Que faire ensuite

Pour que les gestionnaires de files d'attente multi-instances disposent de la plus haute disponibilité, vous devez créer des applications client qui peuvent se reconnecter et des applications serveur qui peuvent être redémarrées ; voir [Récupération d'application](#).

Concepts associés

[Récupération d'application](#)

[reconnexion client automatique](#)

[Reconnexion canal et client](#)

[Gestionnaires de files d'attente multi-instance](#)

[Fichiers de configuration du gestionnaire de files d'attente, qm.ini](#)

[Système de fichiers partagé](#)

[Domaines et gestionnaires de files d'attente multi-instances Windows](#)

Tâches associées

[Sauvegarde des données de gestionnaire de files d'attente](#)

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

[Transfert d'un gestionnaire de files d'attente vers le stockage MSCS](#)

[Vérification du verrouillage du système de fichiers partagé](#)

[Utilisation des services](#)

Référence associée

[amqmfsc \(vérification du système de fichiers\)](#)

Information associée

[Test d'un système de fichiers partagé à des fins de compatibilité avec les gestionnaires de files d'attente multi-instances IBM MQ](#)

[Instruction de test pour les systèmes de fichiers de gestionnaire de files d'attente multi-instance IBM MQ](#)

Multi

Suppression d'un gestionnaire de files d'attente multi-instance

Sur Multiplatforms, pour supprimer complètement un gestionnaire de files d'attente multi-instance, utilisez la commande **dlmqm** pour supprimer le gestionnaire de files d'attente, puis supprimez des instances d'autres serveurs à l'aide des commandes **rmvmqinf** ou **dlmqm**.

Exécutez la commande **dlmqm** pour supprimer un gestionnaire de files d'attente dont des instances sont définies sur d'autres serveurs, sur n'importe quel serveur sur lequel ce gestionnaire de files d'attente est défini. Il n'est pas nécessaire d'exécuter la commande **dlmqm** sur le même serveur que celui sur lequel vous l'avez créée. Exécutez ensuite la commande **rmvmqinf** ou **dlmqm** sur tous les autres serveurs ayant une définition du gestionnaire de files d'attente.

Vous ne pouvez supprimer un gestionnaire de files d'attente que lorsqu'il est arrêté. Au moment de la suppression, aucune instance n'est en cours d'exécution et le gestionnaire de files d'attente, à proprement parler, n'est ni un gestionnaire de files d'attente unique ni un gestionnaire de files d'attente multi-instance ; il s'agit simplement d'un gestionnaire de files d'attente dont les données de gestionnaire de files d'attente et les journaux sont associés à un partage distant. Lorsque vous supprimez un gestionnaire de files d'attente, ses données de gestionnaire de files d'attente et ses journaux sont supprimés et la strophe de gestionnaire de files d'attente est supprimée du fichier `mq5.ini` sur le serveur sur lequel vous avez émis la commande **dlmqm**. Vous devez avoir accès au partage de réseau contenant les données et les journaux du gestionnaire de files d'attente lorsque vous supprimez le gestionnaire de files d'attente.

Sur les autres serveurs sur lesquels vous avez précédemment créé des instances du gestionnaire de files d'attente, il existe également des entrées dans les fichiers `mqs.ini` sur ces serveurs. Vous devez visiter chaque serveur à tour de rôle et supprimer la section du gestionnaire de files d'attente en exécutant la commande **rmvmqinf** *Nom de la section du gestionnaire de files d'attente*.

Linux **AIX** Sur les systèmes AIX and Linux , si vous avez placé un fichier `mqs.ini` commun dans le stockage réseau et que vous l'avez référencé à partir de tous les serveurs en définissant la variable d'environnement `AMQ_MQS_INI_LOCATION` sur chaque serveur, vous devez supprimer le gestionnaire de files d'attente d'un seul de ses serveurs car il n'y a qu'un seul fichier `mqs.ini` à mettre à jour.

Exemple

Premier serveur

```
dltmqm QM1
```

Autres serveurs sur lesquels des instances sont définies

```
rmvmqinf QM1 ou
```

```
dltmqm QM1
```

Linux **AIX** Retour à un gestionnaire de files d'attente mono-instance sous AIX and Linux
Convertissez un gestionnaire de files d'attente multi-instance en un gestionnaire de files d'attente mono-instance sur les AIX and Linux en arrêtant l'instance de secours. Puis, redémarrez l'instance active et ne définissez pas l'option autorisant les instances de secours.

Avant de commencer

Vous disposez d'au moins trois serveurs configurés pour exécuter un gestionnaire de files d'attente en tant que gestionnaire de files d'attente multi-instance. Le gestionnaire de files d'attente est actuellement exécuté en tant que gestionnaire de files d'attente multi-instance, avec une instance de secours active.

Pourquoi et quand exécuter cette tâche

La tâche implique la désactivation de l'instance de secours active de sorte que seul le gestionnaire de file d'attente multi-instance en cours d'exécution reste actif. Pour empêcher le démarrage d'une instance de secours ultérieurement, vous devez arrêter l'instance active, puis la redémarrer. Lorsque vous redémarrez l'instance, vous la redémarrez en tant que gestionnaire de files d'attente mono-instance qui empêche le démarrage des instances de secours. L'arrêt de l'instance de secours est réalisé lors d'une autre étape, vous permettant ainsi de redémarrer l'instance active ultérieurement. Vous pouvez arrêter les deux instances en exécutant la commande `endmqm QMgrName` standard sur le serveur exécutant le gestionnaire de files d'attente actif.

Procédure

1. Arrêtez l'instance de secours du gestionnaire de files d'attente.

Sur le serveur exécutant l'instance de secours :

```
endmqm -w QMgrName
```

2. Arrêtez l'instance active du gestionnaire de files d'attente.

Sur le serveur exécutant l'instance active :

```
endmqm -w (QMgrName)
```

3. Redémarrez le gestionnaire de files d'attente, en empêchant les instances de secours.

Sur le serveur qui va exécuter le gestionnaire de files d'attente :

```
stmqm QMgrName
```

Que faire ensuite

Vous pouvez exécuter le gestionnaire de files d'attente en tant qu'instance unique sur le même serveur que les données du gestionnaire de files d'attente.

Lorsque le gestionnaire de files d'attente est arrêté, transférez de nouveau les données du gestionnaire de files d'attente vers le serveur qui exécute le gestionnaire de files d'attente. Vous pouvez également installer IBM MQ, puis déplacer la définition de configuration du gestionnaire de files d'attente sur le serveur avec les données de gestionnaire de files d'attente. Ces deux tâches sont des variantes des étapes dans «Conversion d'une instance unique en gestionnaire de files d'attente multi-instance sous AIX and Linux», à la page 597 pour la création d'un gestionnaire de files d'attente multi-instance.

Windows Retour à un gestionnaire de files d'attente mono-instance sous Windows

Convertissez un gestionnaire de files d'attente multi-instance en un gestionnaire de files d'attente mono-instance sur les plateformes Windows en arrêtant l'instance de secours. Puis, redémarrez l'instance active et ne définissez pas l'option autorisant les instances de secours.

Avant de commencer

Vous disposez d'au moins trois serveurs configurés pour exécuter un gestionnaire de files d'attente en tant que gestionnaire de files d'attente multi-instance. Le gestionnaire de files d'attente est actuellement exécuté en tant que gestionnaire de files d'attente multi-instance, avec une instance de secours active.

Pourquoi et quand exécuter cette tâche

La tâche implique la désactivation de l'instance de secours active de sorte que seul le gestionnaire de file d'attente multi-instance en cours d'exécution reste actif. Pour empêcher le démarrage d'une instance de secours ultérieurement, vous devez arrêter l'instance active, puis la redémarrer. Lorsque vous redémarrez l'instance, vous la redémarrez en tant que gestionnaire de files d'attente mono-instance qui empêche le démarrage des instances de secours. L'arrêt de l'instance de secours est réalisé lors d'une autre étape, vous permettant ainsi de redémarrer l'instance active ultérieurement. Vous pouvez arrêter les deux instances en exécutant la commande `endmqm QMgrName` standard sur le serveur exécutant le gestionnaire de files d'attente actif.

Procédure

1. Arrêtez l'instance de secours du gestionnaire de files d'attente.

Sur le serveur exécutant l'instance de secours :

```
endmqm -w QMgrName
```

2. Arrêtez l'instance active du gestionnaire de files d'attente.

Sur le serveur exécutant l'instance active :

```
endmqm -w (QMgrName)
```

3. Redémarrez le gestionnaire de files d'attente, en empêchant les instances de secours.

Sur le serveur qui va exécuter le gestionnaire de files d'attente :

```
strmqm QMgrName
```

Que faire ensuite

Vous pouvez exécuter le gestionnaire de files d'attente en tant qu'instance unique sur le même serveur que les données du gestionnaire de files d'attente.

Lorsque le gestionnaire de files d'attente est arrêté, transférez de nouveau les données du gestionnaire de files d'attente vers le serveur qui exécute le gestionnaire de files d'attente. Vous pouvez également installer IBM MQ, puis déplacer la définition de configuration du gestionnaire de files d'attente sur le

serveur avec les données de gestionnaire de files d'attente. Les deux tâches sont des variantes des étapes de la rubrique Conversion d'une instance unique en gestionnaire de files d'attente multi-instance sur Windows.

Multi

Démarrage et arrêt d'un gestionnaire de files d'attente multi-instance

Démarrage et arrêt d'un gestionnaire de files d'attente configuré sur Multiplatforms en tant que gestionnaire de files d'attente mono-instance ou multi-instance.

Une fois que vous avez défini un gestionnaire de files d'attente multi-instance sur une paire de serveurs, vous pouvez exécuter le gestionnaire de files d'attente sur l'un des serveurs, soit en tant que gestionnaire de files d'attente mono-instance, soit en tant que gestionnaire de files d'attente multi-instance.

Pour exécuter un gestionnaire de files d'attente multi-instance, démarrez le gestionnaire de files d'attente sur l'un des serveurs à l'aide de la commande **strmqm -x QM1** ; l'option -x permet à l'instance de basculer. Elle devient l' *instance active*. Démarrez l'instance de secours sur l'autre serveur à l'aide de la même commande **strmqm -x QM1** ; l'option -x permet à l'instance de démarrer en tant qu'instance de secours.

Le gestionnaire de files d'attente est maintenant en cours d'exécution avec une instance active qui traite toutes les demandes et une instance de secours qui est prête à prendre le relais si l'instance active échoue. L'instance active bénéficie d'un accès exclusif aux données et aux journaux du gestionnaire de files d'attente. La base de données de secours attend l'octroi d'un accès exclusif aux données et aux journaux du gestionnaire de files d'attente. Lorsque l'accès exclusif est accordé à la base de données de secours, celle-ci devient l'instance active.

Vous pouvez également basculer manuellement le contrôle vers l'instance de secours en exécutant la commande **endmqm -s** sur l'instance active. La commande **endmqm -s** arrête l'instance active sans arrêter la base de données de secours. Le verrou d'accès exclusif sur les données et les journaux du gestionnaire de files d'attente est libéré et la base de données de secours prend le relais.

Vous pouvez également démarrer et arrêter un gestionnaire de files d'attente configuré avec plusieurs instances sur différents serveurs en tant que gestionnaire de files d'attente d'instance unique. Si vous démarrez le gestionnaire de files d'attente sans utiliser l'option -x dans la commande **strmqm**, les instances du gestionnaire de files d'attente configurées sur d'autres machines ne peuvent pas démarrer en tant qu'instances de secours. Si vous tentez de démarrer une autre instance, vous recevez la réponse indiquant que l'instance de gestionnaire de files d'attente n'est pas autorisée à s'exécuter en tant que instance de secours.

Si vous arrêtez l'instance active d'un gestionnaire de files d'attente multi-instance à l'aide de la commande **endmqm** sans l'option -s, les instances active et de secours s'arrêtent toutes les deux. Si vous arrêtez l'instance de secours à l'aide de la commande **endmqm** avec l'option -x, elle cesse d'être une instance de secours et l'instance active continue de s'exécuter. Vous ne pouvez pas exécuter **endmqm** sans l'option -x sur la base de données de secours.

Seules deux instances de gestionnaire de files d'attente peuvent s'exécuter simultanément ; l'une est l'instance active et l'autre est une instance de secours. Si vous démarrez deux instances en même temps, IBM MQ n'a aucun contrôle sur l'instance qui devient l'instance active ; elle est déterminée par le système de fichiers réseau. La première instance à obtenir un accès exclusif aux données du gestionnaire de files d'attente devient l'instance active.

Remarque : Avant de redémarrer un gestionnaire de files d'attente défaillant, vous devez déconnecter vos applications de cette instance du gestionnaire de files d'attente. Si vous ne le faites pas, le gestionnaire de files d'attente risque de ne pas redémarrer correctement.

Multi

Système de fichiers partagé

Sur les plateformes multiples, un gestionnaire de files d'attente multi-instance utilise un système de fichiers réseau pour gérer les instances de gestionnaire de files d'attente.

Un gestionnaire de files d'attente multi-instance automatise la reprise en ligne à l'aide d'une combinaison de verrous de système de fichiers et de données et de journaux de gestionnaire de files d'attente partagées. Une seule instance d'un gestionnaire de files d'attente peut avoir un accès exclusif aux

données et aux journaux du gestionnaire de files d'attente partagé. Lorsqu'il obtient l'accès, il devient l'instance active. L'autre instance qui ne parvient pas à obtenir un accès exclusif attend en tant qu'instance de secours jusqu'à ce que les données et les journaux du gestionnaire de files d'attente soient disponibles.

Le système de fichiers en réseau est chargé de libérer les verrous qu'il détient pour l'instance de gestionnaire de files d'attente active. Si l'instance active échoue d'une manière ou d'une autre, le système de fichiers en réseau libère les verrous qu'elle détient pour l'instance active. Dès que le verrou exclusif est libéré, un gestionnaire de files d'attente de secours attend que le verrou tente de l'acquérir. S'il réussit, il devient l'instance active et dispose d'un accès exclusif aux données et aux journaux du gestionnaire de files d'attente sur le système de fichiers partagé. Il continue ensuite à démarrer.

La rubrique connexe, [Planification de la prise en charge du système de fichiers](#), explique comment configurer et vérifier que votre système de fichiers prend en charge les gestionnaires de files d'attente multi-instance.

Un gestionnaire de files d'attente multi-instance ne vous protège pas contre un incident dans le système de fichiers. Il existe plusieurs façons de protéger vos données.

- Investissez dans un stockage fiable, tel que des grappes de disques redondantes (RAID), et incluez-les dans un système de fichiers en réseau doté d'une résilience réseau.
- Sauvegardez les journaux linéaires IBM MQ sur un autre support et, en cas de défaillance de votre support de journal principal, effectuez une reprise à l'aide des journaux sur le support de remplacement. Vous pouvez utiliser un gestionnaire de files d'attente de sauvegarde pour administrer ce processus.

Multi *Plusieurs instances de gestionnaire de files d'attente*

Un gestionnaire de files d'attente multi-instance est résilient car il utilise une instance de gestionnaire de files d'attente de secours pour restaurer la disponibilité du gestionnaire de files d'attente après une défaillance.

La réplication des instances de gestionnaire de files d'attente est un moyen très efficace d'améliorer la disponibilité des processus de gestionnaire de files d'attente. A l'aide d'un modèle de disponibilité simple, uniquement à des fins d'illustration: si la fiabilité d'une instance d'un gestionnaire de files d'attente est de 99% (sur une année, le temps d'indisponibilité cumulé est de 3.65 jours), l'ajout d'une autre instance du gestionnaire de files d'attente augmente la disponibilité à 99.99% (sur une année, le temps d'indisponibilité cumulé est d'environ une heure).

Il s'agit d'un modèle trop simple pour vous donner des estimations numériques pratiques de la disponibilité. Pour modéliser la disponibilité de manière réaliste, vous devez collecter des statistiques sur le temps moyen entre les pannes (MTBF) et le temps moyen de réparation (MTTR), ainsi que sur la distribution de probabilité du temps entre les pannes et les temps de réparation.

Le terme gestionnaire de files d'attente multi-instance fait référence à la combinaison des instances actives et de secours du gestionnaire de files d'attente qui partagent les données et les journaux du gestionnaire de files d'attente. Les gestionnaires de files d'attente multi-instance vous protègent contre l'échec des processus de gestionnaire de files d'attente en ayant une instance du gestionnaire de files d'attente active sur un serveur et une autre instance du gestionnaire de files d'attente en veille sur un autre serveur, prête à prendre le relais automatiquement en cas d'échec de l'instance active.

Multi *Basculement ou basculement*

Une instance de gestionnaire de files d'attente de secours prend le relais de l'instance active à la demande (commutation) ou en cas de défaillance de l'instance active (reprise en ligne).

- La *commutation* a lieu lorsqu'une instance de secours démarre en réponse à l'émission de la commande **endmqm -s** sur l'instance de gestionnaire de files d'attente active. Vous pouvez spécifier les paramètres **endmqm -c**, **-i** ou **-p** pour contrôler l'arrêt brutal du gestionnaire de files d'attente.

Remarque : La commutation n'a lieu que si une instance de gestionnaire de files d'attente de secours est déjà démarrée. La commande **endmqm -s** libère le verrou du gestionnaire de files d'attente actif et autorise la commutation: elle ne démarre pas d'instance de gestionnaire de files d'attente de secours.

- La *reprise en ligne* se produit lorsque le verrouillage des données du gestionnaire de files d'attente détenues par l'instance active est libéré car l'instance semblait s'arrêter de manière inattendue (c'est-à-dire sans qu'une commande **endmqm** soit émise).

Lorsque l'instance de secours prend le relais en tant qu'instance active, elle écrit un message dans le journal des erreurs du gestionnaire de files d'attente.

Les clients reconnectables sont automatiquement reconnectés en cas de défaillance ou de basculement d'un gestionnaire de files d'attente. Il n'est pas nécessaire d'inclure l'indicateur `-r` dans la commande **endmqm** pour demander la reconnexion du client. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java.

Si vous constatez que vous ne pouvez pas redémarrer une instance ayant échoué, même si la reprise en ligne a eu lieu et que l'instance de secours est devenue active, vérifiez si les applications connectées en local à l'instance ayant échoué se sont déconnectées de l'instance ayant échoué.

Les applications connectées en local doivent s'arrêter ou se déconnecter d'une instance de gestionnaire de files d'attente en échec pour que l'instance en échec puisse être redémarrée. Toutes les applications connectées en local à l'aide de liaisons partagées (qui est le paramètre par défaut) qui maintiennent une connexion à une instance ayant échoué agissent pour empêcher le redémarrage de l'instance.

S'il n'est pas possible d'arrêter les applications connectées en local ou de s'assurer qu'elles se déconnectent lorsque l'instance du gestionnaire de files d'attente local échoue, envisagez d'utiliser des liaisons isolées. Les applications connectées en local à l'aide de liaisons isolées n'empêchent pas le redémarrage de l'instance de gestionnaire de files d'attente locale, même si elles ne se déconnectent pas.

Multi **Reconnexion canal et client**

La reconnexion du canal et du client est un élément essentiel de la restauration du traitement des messages une fois qu'une instance de gestionnaire de files d'attente de secours est devenue active.

Les instances de gestionnaire de files d'attente multi-instance sont installées sur des serveurs avec des adresses réseau différentes. Vous devez configurer les canaux et les clients IBM MQ avec des informations de connexion pour toutes les instances de gestionnaire de files d'attente. Lorsqu'une instance de secours prend le relais, les clients et les canaux sont automatiquement reconnectés à l'instance de gestionnaire de files d'attente nouvellement active à la nouvelle adresse réseau. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java.

La conception est différente de celle des environnements à haute disponibilité tels que HA-CMP. HA-CMP fournit une adresse IP virtuelle pour le cluster et transfère l'adresse au serveur actif. La reconnexion IBM MQ ne modifie pas ou ne redirige pas les adresses IP. Il fonctionne en se reconnectant à l'aide des adresses réseau que vous avez définies dans les définitions de canal et les connexions client. En tant qu'administrateur, vous devez définir les adresses réseau dans les définitions de canal et les connexions client à toutes les instances d'un gestionnaire de files d'attente multi-instance. La meilleure façon de configurer des adresses réseau pour un gestionnaire de files d'attente multi-instance dépend de la connexion:

Canaux de gestionnaire de file d'attente

L'attribut **CONNNAME** des canaux est une liste de noms de connexion séparés par des virgules ; par exemple, **CONNNAME ('127.0.0.1(1234), 192.0.2.0(4321)')**. Les connexions sont tentées dans l'ordre indiqué dans la liste des connexions jusqu'à ce qu'une connexion soit établie. Si aucune connexion n'aboutit, le canal tente de se reconnecter.

Canaux de cluster

En règle générale, aucune configuration supplémentaire n'est requise pour que les gestionnaires de files d'attente multi-instance fonctionnent dans un cluster.

Si un gestionnaire de files d'attente se connecte à un gestionnaire de files d'attente de référentiel, ce dernier reconnaît l'adresse réseau du gestionnaire de files d'attente. Il fait référence à **CONNNAME** du canal **CLUSRCVR** sur le gestionnaire de files d'attente. Sur **TCPIP**, le gestionnaire de files d'attente définit automatiquement **CONNNAME** si vous l'omettez ou le configurez à blanc. Lorsqu'une instance de secours prend le relais, son adresse IP remplace l'adresse IP de l'instance active précédente en tant que **CONNNAME**.

Si nécessaire, vous pouvez configurer manuellement CONNAME avec la liste des adresses réseau des instances de gestionnaire de files d'attente.

Connexions client

Les connexions client peuvent utiliser des listes de connexions ou des groupes de gestionnaires de files d'attente pour sélectionner d'autres connexions.

Lorsque la reprise en ligne se produit, la reconnexion prend un certain temps. Le gestionnaire de files d'attente de secours doit terminer son démarrage. Les clients qui ont été connectés au gestionnaire de files d'attente défaillant doivent détecter l'échec de la connexion et démarrer une nouvelle connexion client. Si une nouvelle connexion client sélectionne le gestionnaire de files d'attente de secours qui est devenu récemment actif, le client est reconnecté au même gestionnaire de files d'attente.

Si le client se trouve au milieu d'un appel MQI lors de la reconnexion, il doit tolérer une attente étendue avant la fin de l'appel.

Si l'échec se produit lors d'un transfert par lots sur un canal de transmission de messages, le lot est annulé et redémarré.

Le basculement est plus rapide que le basculement et ne prend que le temps d'arrêter une instance du gestionnaire de files d'attente et d'en démarrer une autre. Pour un gestionnaire de files d'attente avec peu d'enregistrements de journal à réexécuter, la commutation peut au mieux prendre de l'ordre de quelques secondes. Pour estimer la durée de la reprise en ligne, vous devez ajouter le temps nécessaire à la détection de l'incident. Au mieux, la détection prend de l'ordre de 10 secondes et peut durer plusieurs minutes, en fonction du réseau et du système de fichiers.

Multi Récupération d'application

La reprise d'application est la poursuite automatisée du traitement de l'application après la reprise en ligne. La reprise d'application après la reprise en ligne nécessite une conception minutieuse. Certaines applications doivent être conscientes que la reprise en ligne a eu lieu.

L'objectif de la reprise de l'application est de permettre à l'application de continuer le traitement avec un bref délai. Avant de poursuivre le nouveau traitement, l'application doit revenir en arrière et soumettre à nouveau l'unité de travail qu'elle traitait lors de l'échec.

Un problème pour la reprise d'application est la perte du contexte partagé entre IBM MQ MQI client et le gestionnaire de files d'attente et stocké dans le gestionnaire de files d'attente. IBM MQ MQI client restaure la majeure partie du contexte, mais certaines parties du contexte ne peuvent pas être restaurées de manière fiable. Les sections suivantes décrivent certaines propriétés de reprise d'application et comment elles affectent la reprise des applications connectées à un gestionnaire de files d'attente multi-instance.

messagerie transactionnelle

Du point de vue de la distribution des messages, la reprise en ligne ne modifie pas les propriétés persistantes de la messagerie IBM MQ . Si les messages sont persistants et correctement gérés dans les unités de travail, ils ne sont pas perdus lors d'une reprise en ligne.

Du point de vue du traitement des transactions, les transactions sont annulées ou validées après la reprise en ligne.

Les transactions non validées sont annulées. Après la reprise en ligne, une application reconnectable reçoit un code anomalie MQRC_BACKED_OUT pour indiquer que la transaction a échoué. Il doit ensuite redémarrer la transaction.

Les transactions validées sont des transactions qui ont atteint la deuxième phase d'une validation en deux phases ou des transactions en une phase (message uniquement) qui ont commencé MQCMIT.

Si le gestionnaire de files d'attente est le coordinateur de transactions et que MQCMIT a commencé la deuxième phase de sa validation en deux phases avant l'échec, la transaction aboutit. L'exécution est sous le contrôle du gestionnaire de files d'attente et se poursuit lorsque le gestionnaire de files d'attente est à nouveau en cours d'exécution. Dans une application reconnectable, l'appel MQCMIT se termine normalement.

Dans une validation en une seule phase, qui implique uniquement des messages, une transaction qui a démarré le traitement de la validation se termine normalement sous le contrôle du gestionnaire de files d'attente une fois qu'elle est à nouveau en cours d'exécution. Dans une application reconnectable, MQCMIT se termine normalement.

Les clients reconnectables peuvent utiliser des transactions monophasées sous le contrôle du gestionnaire de files d'attente en tant que coordinateur de transactions. Le client transactionnel étendu ne prend pas en charge la reconnexion. Si une reconnexion est demandée lorsque le client transactionnel se connecte, la connexion aboutit, mais sans possibilité de reconnexion. La connexion se comporte comme si elle n'était pas reconnectable.

Redémarrage ou reprise de l'application

La reprise en ligne interrompt une application. Après un échec, une application peut redémarrer depuis le début ou reprendre le traitement après l'interruption. Cette dernière est appelée *reconnexion automatique du client*. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java.

Avec une application IBM MQ MQI client, vous pouvez définir une option de connexion pour reconnecter automatiquement le client. Les options sont MQCNO_RECONNECT ou MQCNO_RECONNECT_Q_MGR. Si aucune option n'est définie, le client ne tente pas de se reconnecter automatiquement et l'échec du gestionnaire de files d'attente renvoie MQRC_CONNECTION_BROKEN au client. Vous pouvez concevoir le client pour qu'il tente de démarrer une nouvelle connexion en émettant un nouvel appel MQCONN ou MQCONNX.

Les programmes serveur doivent être redémarrés ; ils ne peuvent pas être reconnectés automatiquement par le gestionnaire de files d'attente au moment où ils étaient en cours de traitement lorsque le gestionnaire de files d'attente ou le serveur a échoué. Les programmes serveur IBM MQ ne sont généralement pas redémarrés sur l'instance de gestionnaire de files d'attente de secours en cas d'échec d'une instance de gestionnaire de files d'attente multi-instance.

Vous pouvez automatiser un programme serveur IBM MQ pour qu'il redémarre sur le serveur de secours de deux manières:

1. Conditionnez votre application serveur en tant que service de gestionnaire de files d'attente. Il est redémarré lorsque le gestionnaire de files d'attente de secours redémarre.
2. Ecrivez votre propre logique de basculement, déclenchée par exemple, par le message de journal de basculement écrit par une instance de gestionnaire de files d'attente de secours au démarrage. L'instance d'application doit ensuite appeler MQCONN ou MQCONNX après son démarrage pour créer une connexion au gestionnaire de files d'attente.

Détection du basculement

Certaines applications doivent être conscientes de la reprise en ligne, d'autres non. Prenez ces deux exemples.

1. Une application de messagerie qui extrait ou reçoit des messages via un canal de messagerie ne nécessite normalement pas que le gestionnaire de files d'attente à l'autre extrémité du canal soit en cours d'exécution: il est peu probable qu'il soit affecté si le gestionnaire de files d'attente à l'autre extrémité du canal redémarre sur une instance de secours.
2. Une application IBM MQ MQI client traite les entrées de messages persistants d'une file d'attente et place les réponses de messages persistants dans une autre file d'attente dans le cadre d'une unité de travail unique: si elle gère un code anomalie MQRC_BACKED_OUT provenant de MQPUT, MQGET ou MQCMIT dans le point de synchronisation en redémarrant l'unité de travail, aucun message n'est perdu. De plus, l'application n'a pas besoin d'effectuer de traitement spécial pour traiter un échec de connexion.

Supposons toutefois, dans le deuxième exemple, que l'application parcourt la file d'attente pour sélectionner le message à traiter à l'aide de l'option MQGET, MQGMO_MSG_UNDER_CURSOR. La reconnexion réinitialise le curseur de navigation et l'appel MQGET ne renvoie pas le message correct.

Dans cet exemple, l'application doit prendre en compte la reprise en ligne qui s'est produite. De plus, avant d'émettre un autre MQGET pour le message sous le curseur, l'application doit restaurer le curseur de navigation.

La perte du curseur de navigation est un exemple de la façon dont le contexte d'application change après la reconnexion. D'autres cas sont documentés dans [«Récupération d'un client reconnecté automatiquement»](#), à la page 611.

Vous disposez de trois modèles de conception alternatifs pour les applications IBM MQ MQI client après la reprise en ligne. Un seul d'entre eux n'a pas besoin de détecter la reprise en ligne.

Aucune reconnexion

Dans ce modèle, l'application arrête tout traitement sur la connexion en cours lorsque la connexion est interrompue. Pour que l'application puisse continuer le traitement, elle doit établir une nouvelle connexion avec le gestionnaire de files d'attente. L'application est entièrement responsable du transfert des informations d'état dont elle a besoin pour continuer le traitement sur la nouvelle connexion. Les applications client existantes qui se reconnectent à un gestionnaire de files d'attente après la perte de leur connexion sont écrites de cette manière.

Le client reçoit un code anomalie, tel que MQRC_CONNECTION_BROKEN ou MQRC_Q_MGR_NOT_AVAILABLE, du prochain appel MQI après la perte de la connexion. L'application doit supprimer toutes ses informations d'état IBM MQ, telles que les descripteurs de file d'attente, et émettre un nouvel appel MQCONN ou MQCONNX pour établir une nouvelle connexion, puis rouvrir les objets IBM MQ qu'elle doit traiter.

Le comportement MQI par défaut consiste à rendre le descripteur de connexion du gestionnaire de files d'attente inutilisable après la perte d'une connexion au gestionnaire de files d'attente. La valeur par défaut est équivalente à la définition de l'option MQCNO_RECONNECT_DISABLED sur MQCONNX pour empêcher la reconnexion de l'application après la reprise en ligne.

Tolérant aux pannes

Ecrivez l'application de sorte qu'elle ne soit pas affectée par la reprise en ligne. Parfois, le traitement des erreurs est suffisant pour traiter la reprise en ligne.

Prise en compte de la reconnexion

Enregistrez un gestionnaire d'événements MQCBT_EVENT_HANDLER auprès du gestionnaire de files d'attente. Le gestionnaire d'événements est envoyé avec MQRC_RECONNECTING lorsque le client tente de se reconnecter au serveur et MQRC_RECONNECTED après une reconnexion réussie. Vous pouvez ensuite exécuter une routine pour rétablir un état prévisible afin que l'application client puisse poursuivre le traitement.

Récupération d'un client reconnecté automatiquement

La reprise en ligne est un événement inattendu, et pour qu'un client reconnecté automatiquement fonctionne comme prévu, les conséquences de la reconnexion doivent être prévisibles.

L'utilisation des transactions est un élément majeur de la transformation d'un échec inattendu en une reprise prévisible et fiable.

Dans la section précédente, un exemple, [«2»](#), à la page 610, a été donné d'une IBM MQ MQI client utilisant une transaction locale pour coordonner MQGET et MQPUT. Le client émet un appel MQCMIT ou MQBACK en réponse à une erreur MQRC_BACKED_OUT, puis soumet à nouveau la transaction annulée. La défaillance du gestionnaire de files d'attente entraîne l'annulation de la transaction et le comportement de l'application client garantit qu'aucune transaction et aucun message ne sont perdus.

Notez que pour un rappel, il peut être nécessaire de reprendre l'application consommatrice si l'état du paramètre de consommateur de rappel est: MQCS_SUSPENDED_USER_ACTION.

Tous les états de programme ne sont pas gérés dans le cadre d'une transaction et, par conséquent, les conséquences de la reconnexion deviennent plus difficiles à comprendre. Vous devez savoir comment la reconnexion modifie l'état d'une IBM MQ MQI client afin de concevoir votre application client pour survivre à la reprise en ligne du gestionnaire de files d'attente.

Vous pouvez décider de concevoir votre application sans code de reprise en ligne spécial, en traitant les erreurs de reconnexion avec la même logique que les autres erreurs. Vous pouvez également choisir de reconnaître que la reconnexion nécessite un traitement d'erreur spécial et d'enregistrer un gestionnaire d'événements auprès de IBM MQ afin d'exécuter une routine pour gérer la reprise en ligne. La routine peut traiter elle-même le traitement de la reconnexion ou définir un indicateur pour indiquer à l'unité d'exécution du programme principal que lorsqu'elle reprend le traitement, elle doit effectuer le traitement de la reprise.

L'environnement IBM MQ MQI client est conscient de la reprise en ligne elle-même et restaure autant de contexte qu'il peut, après une reconnexion, en stockant certaines informations d'état dans le client et en émettant des appels MQI supplémentaires pour le compte de l'application client afin de restaurer son état IBM MQ . Par exemple, les descripteurs des objets qui étaient ouverts au point de défaillance sont restaurés et les files d'attente dynamiques temporaires sont ouvertes avec le même nom. Mais il y a des changements qui sont inévitables et vous avez besoin de votre conception pour faire face à ces changements. Les changements peuvent être classés en cinq catégories:

1. Les nouvelles erreurs ou les erreurs non diagnostiquées précédemment sont renvoyées par les appels MQI jusqu'à ce qu'un nouvel état de contexte cohérent soit restauré par le programme d'application.

Un exemple de réception d'une nouvelle erreur est le code retour MQRC_CONTEXT_NOT_AVAILABLE lors de la tentative de transmission du contexte après la sauvegarde du contexte avant la reconnexion. Le contexte ne peut pas être restauré après la reconnexion car le contexte de sécurité n'est pas transmis à un programme client non autorisé. Cela permettrait à un programme d'application malveillant d'obtenir le contexte de sécurité.

En règle générale, les applications traitent les erreurs courantes et prévisibles d'une manière soigneusement conçue et relègue les erreurs rares à un gestionnaire d'erreurs générique. Le gestionnaire d'erreurs peut se déconnecter de IBM MQ et se reconnecter à nouveau, voire arrêter complètement le programme. Pour améliorer la continuité, vous devrez peut-être traiter certaines erreurs d'une manière différente.

2. Des messages non persistants peuvent être perdus.
3. Les transactions sont annulées (ce qui peut également interrompre les consommateurs asynchrones, voir le texte précédent).
4. Les appels MQGET ou MQPUT utilisés en dehors d'un point de synchronisation peuvent être interrompus par la perte possible d'un message.
5. La temporisation a provoqué des erreurs, en raison d'une attente prolongée dans un appel MQI.

Certains détails sur le contexte perdu sont répertoriés dans la section suivante.

- Les messages non persistants sont supprimés, sauf s'ils sont placés dans une file d'attente avec l'option NPMCLASS (HIGH) et que l'échec du gestionnaire de files d'attente n'a pas interrompu l'option de stockage des messages non persistants à l'arrêt.
- Un abonnement non durable est perdu lorsqu'une connexion est interrompue. Lors de la reconnexion, il est rétabli. Envisagez d'utiliser un abonnement durable.
- L'intervalle get-wait est recalculé ; si sa limite est dépassée, il renvoie MQRC_NO_MSG_AVAILABLE. De même, l'expiration de l'abonnement est recalculée pour donner le même délai d'expiration global.
- La position du curseur de navigation dans une file d'attente est perdue ; elle est généralement rétablie avant le premier message.
 - Les appels MQGET qui spécifient MQGMO_BROWSE_MSG_UNDER_CURSOR ou MQGMO_MSG_UNDER_CURSOR échouent avec le code anomalie MQRC_NO_MSG_AVAILABLE.
 - Les messages verrouillés pour la navigation sont déverrouillés.
 - L'exploration des messages marqués avec la portée de descripteur n'est pas marquée et peut être parcourue à nouveau.
 - Dans la plupart des cas, les messages marqués par une navigation coopérative ne sont pas marqués.

- Le contexte de sécurité est perdu. Les tentatives d'utilisation du contexte de message sauvegardé, telles que l'insertion d'un message avec MQPMO_PASS_ALL_CONTEXT , échouent avec MQRC_CONTEXT_NOT_AVAILABLE.
- Les jetons de message sont perdus. MQGET à l'aide d'un jeton de message renvoie le code anomalie MQRC_NO_MSG_AVAILABLE.

Remarque : *MsgId* et *CorrelId*, car ils font partie du message, sont conservés avec le message lors de la reprise en ligne, de sorte que MQGET l'utilisation de *MsgId* ou *CorrelId* fonctionne comme prévu.

- Les messages placés dans une file d'attente sous un point de synchronisation dans une transaction non validée ne sont plus disponibles.
- Le traitement des messages dans un ordre logique ou dans un groupe de messages génère le code retour MQRC_RECONNECT_INCOMPATIBLE après la reconnexion.
- Un appel MQI peut renvoyer MQRC_RECONNECT_FAILED plutôt que le MQRC_CONNECTION_BROKEN plus général que les clients reçoivent généralement aujourd'hui.
- La reconnexion lors d'un MQPUT appel en dehors du point de synchronisation renvoie MQRC_CALL_INTERRUPTED si IBM MQ MQI client ne sait pas si le message a été correctement distribué au gestionnaire de files d'attente. La reconnexion pendant MQCMIT se comporte de la même manière.
- MQRC_CALL_INTERRUPTED est renvoyé-après une reconnexion réussie-si IBM MQ MQI client n'a reçu aucune réponse du gestionnaire de files d'attente indiquant la réussite ou l'échec de
 - la distribution d'un message persistant à l'aide d'un appel MQPUT en dehors du point de synchronisation.
 - la distribution d'un message persistant ou d'un message avec persistance par défaut à l'aide d'un appel MQPUT1 en dehors du point de synchronisation.
 - la validation d'une transaction à l'aide d'un appel MQCMIT. La réponse n'est renvoyée qu'après une reconnexion réussie.
- Les canaux sont redémarrés en tant que nouvelles instances (il peut également s'agir de canaux différents), de sorte qu'aucun état de sortie de canal n'est conservé.
- Les files d'attente dynamiques temporaires sont restaurées dans le cadre du processus de récupération des clients reconnectables dont les files d'attente dynamiques temporaires étaient ouvertes. Aucun message d'une file d'attente dynamique temporaire n'est restauré, mais les applications dont la file d'attente a été ouverte ou dont le nom a été mémorisé peuvent continuer le traitement.

Il est possible que si la file d'attente est utilisée par une application autre que celle qui l'a créée, elle ne soit pas restaurée assez rapidement pour être présente lors de sa prochaine référence. Par exemple, si un client crée une file d'attente dynamique temporaire en tant que file d'attente de réponse et qu'un message de réponse doit être placé dans la file d'attente par un canal, la file d'attente peut ne pas être récupérée à temps. Dans ce cas, le canal place généralement le message de réponse dans la file d'attente des messages non livrés.

Si une application client reconnectable ouvre une file d'attente dynamique temporaire par son nom (car une autre application l'a déjà créée), lorsque la reconnexion se produit, IBM MQ MQI client ne peut pas recréer la file d'attente dynamique temporaire car il ne dispose pas du modèle à partir duquel la créer. Dans l'interface MQI, une seule application peut ouvrir la file d'attente dynamique temporaire par modèle. Les autres applications qui souhaitent utiliser la file d'attente dynamique temporaire doivent utiliser MQPUT1 ou des liaisons de serveur, ou être en mesure de relancer la reconnexion en cas d'échec.

Seuls les messages non persistants peuvent être placés dans une file d'attente dynamique temporaire et ces messages sont perdus lors de la reprise en ligne ; cette perte est vraie pour les messages placés dans une file d'attente dynamique temporaire à l'aide de MQPUT1 lors de la reconnexion. Si la reprise en ligne se produit pendant MQPUT1, il se peut que le message ne soit pas inséré, bien que MQPUT1 aboutisse. Une solution palliative à ce problème consiste à utiliser des files d'attente dynamiques permanentes. Toute application de liaisons de serveur peut ouvrir la file d'attente dynamique temporaire par son nom car elle ne peut pas être reconnectée.

Les solutions à haute disponibilité utilisant des gestionnaires de files d'attente multi-instance doivent inclure un mécanisme permettant de récupérer des données après une panne de stockage.

Un gestionnaire de files d'attente multi-instance augmente la disponibilité des processus de gestionnaire de files d'attente, mais pas celle des autres composants, tels que le système de fichiers, utilisés par le gestionnaire de files d'attente pour stocker des messages et d'autres informations.

Une façon de rendre les données hautement disponibles consiste à utiliser un stockage de données résilient en réseau. Vous pouvez soit créer votre propre solution à l'aide d'un système de fichiers en réseau et d'un stockage de données résilient, soit acheter une solution intégrée. Si vous souhaitez combiner la résilience avec la reprise après incident, la réplication de disque asynchrone, qui permet la réplication de disque sur des dizaines, voire des centaines de kilomètres, est disponible.

Vous pouvez configurer la manière dont les différents répertoires IBM MQ sont mappés aux supports de stockage afin d'optimiser l'utilisation de ces derniers. Pour les gestionnaires de files d'attente *multi-instance*, il existe une distinction importante entre deux types de répertoires et de fichiers IBM MQ.

Répertoires qui doivent être partagés entre les instances d'un gestionnaire de files d'attente.

Les informations qui doivent être partagées entre les différentes instances d'un gestionnaire de files d'attente se trouvent dans deux répertoires: les répertoires `qmgrs` et `logs`. Les répertoires doivent se trouver sur un système de fichiers en réseau partagé. Il est conseillé d'utiliser un support de stockage qui offre une haute disponibilité continue et d'excellentes performances car les données changent constamment à mesure que des messages sont créés et supprimés.

Les répertoires et les fichiers qui ne sont pas *ont* à partager entre les instances d'un gestionnaire de files d'attente.

Certains autres répertoires n'ont pas besoin d'être partagés entre différentes instances d'un gestionnaire de files d'attente et sont rapidement restaurés par d'autres moyens que l'utilisation d'un système de fichiers miroir.

- Fichiers exécutables IBM MQ et répertoire des outils. Remplacez-le par une réinstallation ou par une sauvegarde et une restauration à partir d'une archive de fichier sauvegardée.
- Informations de configuration modifiées pour l'installation dans son ensemble. Les informations de configuration sont gérées par IBM MQ, comme le fichier `mqs.ini` sur les systèmes AIX, Linux, and Windows, ou font partie de votre propre gestion des configurations, comme les scripts de configuration **MQSC**. Effectuez une sauvegarde et une restauration à l'aide d'une archive de fichier.
- Sortie à l'échelle de l'installation, telle que les traces, les journaux d'erreurs et les fichiers FFDC. Les fichiers sont stockés dans les sous-répertoires `errors` et `trace` du répertoire de données par défaut. Le répertoire de données par défaut sur les systèmes AIX and Linux est `/var/mqm`. Sous Windows, le répertoire de données par défaut est le répertoire d'installation IBM MQ.

Vous pouvez également utiliser un gestionnaire de files d'attente de sauvegarde pour effectuer des sauvegardes de support régulières d'un gestionnaire de files d'attente multi-instance à l'aide de la journalisation linéaire. Un gestionnaire de files d'attente de sauvegarde ne fournit pas de reprise aussi rapide qu'à partir d'un système de fichiers miroir et ne récupère pas les modifications depuis la dernière sauvegarde. Le mécanisme de gestionnaire de files d'attente de sauvegarde est plus approprié pour une utilisation dans des scénarios de reprise après incident hors site que la récupération d'un gestionnaire de files d'attente après une panne de stockage localisée.

Combinaison de solutions IBM MQ Availability

Les applications utilisent d'autres fonctions IBM MQ pour améliorer la disponibilité. Les gestionnaires de files d'attente multi-instance complètent d'autres fonctions de haute disponibilité.

IBM MQ Les clusters augmentent la disponibilité des files d'attente

Vous pouvez augmenter la disponibilité de la file d'attente en créant plusieurs définitions d'une file d'attente de cluster, jusqu'à une de chaque file d'attente sur chaque gestionnaire du cluster.

Supposons qu'un membre du cluster échoue et qu'un nouveau message soit envoyé à une file d'attente de cluster. Sauf si le message *doit* être envoyé au gestionnaire de files d'attente qui a échoué, le message est envoyé à un autre gestionnaire de files d'attente en cours d'exécution dans le cluster qui possède une définition de la file d'attente.

Bien que les clusters augmentent considérablement la disponibilité, il existe deux scénarios d'échec connexes qui entraînent des retards dans les messages. La génération d'un cluster avec des gestionnaires de files d'attente multi-instance réduit les risques de retard d'un message.

Messages non segmentés

Si un gestionnaire de files d'attente du cluster tombe en panne, aucun autre message pouvant être acheminé vers d'autres gestionnaires de files d'attente du cluster n'est acheminé vers le gestionnaire de files d'attente défaillant. Les messages qui ont déjà été envoyés sont stockés jusqu'à ce que le gestionnaire de files d'attente défaillant soit redémarré.

Affinités

L'affinité est le terme utilisé pour décrire les informations partagées entre deux calculs distincts. Par exemple, il existe une affinité entre une application qui envoie un message de demande à un serveur et la même application qui attend de traiter la réponse. Un autre exemple serait une séquence de messages, le traitement de chaque message dépendant des messages précédents.

Si vous envoyez des messages à des files d'attente en cluster, vous devez prendre en compte les affinités. Avez-vous besoin d'envoyer des messages successifs au même gestionnaire de files d'attente ou chaque message peut-il être envoyé à un membre du cluster?

Si vous devez envoyer des messages au même gestionnaire de files d'attente dans le cluster et qu'il échoue, les messages attendent dans la file d'attente de transmission de l'émetteur jusqu'à ce que le gestionnaire de files d'attente de cluster défaillant soit de nouveau en cours d'exécution.

Si le cluster est configuré avec des gestionnaires de files d'attente multi-instance, le délai d'attente du redémarrage du gestionnaire de files d'attente défaillant est limité à environ une minute pendant la reprise de la base de données de secours. Lorsque la base de données de secours est en cours d'exécution, les messages désactivés reprennent le traitement, les canaux vers l'instance de gestionnaire de files d'attente nouvellement activée sont démarrés et les messages en attente dans les files d'attente de transmission commencent à circuler.

Un moyen possible de configurer un cluster pour éviter que des messages soient retardés par un gestionnaire de files d'attente défaillant consiste à déployer deux gestionnaires de files d'attente différents sur chaque serveur du cluster et à faire en sorte que l'un soit l'instance active et l'autre l'instance de secours des différents gestionnaires de files d'attente. Il s'agit d'une configuration active-de secours qui augmente la disponibilité du cluster.

En plus d'avoir les avantages d'une administration réduite et d'une évolutivité accrue, les clusters continuent de fournir des éléments supplémentaires de disponibilité pour compléter les gestionnaires de files d'attente multi-instance. Les clusters protègent contre d'autres types de défaillance qui affectent à la fois les instances actives et de secours d'un gestionnaire de files d'attente.

Service ininterrompu

Un cluster fournit un service ininterrompu. Les nouveaux messages reçus par le cluster sont envoyés aux gestionnaires de files d'attente actifs pour être traités. Ne vous fiez pas à un gestionnaire de files d'attente multi-instance pour fournir un service ininterrompu car il faut du temps au gestionnaire de files d'attente de secours pour détecter l'échec et terminer son démarrage, pour que ses canaux soient reconnectés et pour que les lots de messages ayant échoué soient soumis à nouveau.

Indisponibilité localisée


Il existe des limitations pratiques sur la distance entre les serveurs actifs, de secours et de système de fichiers, car ils doivent interagir à des vitesses à la milliseconde pour offrir des performances acceptables.

Les gestionnaires de files d'attente en cluster nécessitent des vitesses d'interaction de l'ordre de plusieurs secondes et peuvent être dispersés géographiquement n'importe où dans le monde.

Erreur opérationnelle

En utilisant deux mécanismes différents pour augmenter la disponibilité, vous réduisez les risques qu'une erreur opérationnelle, telle qu'une erreur humaine, compromette vos efforts de disponibilité.

Les groupes de partage de files d'attente augmentent la disponibilité du traitement des messages

 Les groupes de partage de files d'attente, fournis uniquement sous z/OS, permettent à un groupe de gestionnaires de files d'attente de partager le service d'une file d'attente. Si un gestionnaire de files d'attente est défaillant, les autres gestionnaires de files d'attente continuent de traiter tous les messages de la file d'attente. Les gestionnaires de files d'attente multi-instance ne sont pas pris en charge sur z/OS et ne complètent les groupes de partage de files d'attente que dans le cadre d'une architecture de messagerie plus large.

IBM MQ Les clients augmentent la disponibilité des applications

Les programmes IBM MQ MQI client peuvent se connecter à différents gestionnaires de files d'attente dans un groupe de gestionnaires de files d'attente en fonction de la disponibilité du gestionnaire de files d'attente, des pondérations de connexion et des affinités. En exécutant une application sur une machine différente de celle sur laquelle le gestionnaire de files d'attente est en cours d'exécution, vous pouvez améliorer la disponibilité globale d'une solution tant qu'il existe un moyen de reconnecter l'application en cas de défaillance de l'instance de gestionnaire de files d'attente à laquelle elle est connectée.

Les groupes de gestionnaires de files d'attente sont utilisés pour augmenter la disponibilité du client en découplant un client d'un gestionnaire de files d'attente arrêté et en équilibrant la charge des connexions client dans un groupe de gestionnaires de files d'attente, comme un diffuseur IP. L'application client ne doit pas avoir d'affinités avec le gestionnaire de files d'attente défaillant, par exemple une dépendance sur une file d'attente particulière, ou elle ne peut pas reprendre le traitement.

La reconnexion automatique du client et les gestionnaires de files d'attente multi-instance augmentent la disponibilité du client en résolvant certains problèmes d'affinité. La reconnexion client automatique n'est pas prise en charge par IBM MQ classes for Java.

Vous pouvez définir l'option MQCNO MQCNO_RECONNECT_Q_MGR pour forcer un client à se reconnecter au même gestionnaire de files d'attente:

1. Si le gestionnaire de files d'attente d'instance unique précédemment connecté n'est pas en cours d'exécution, une nouvelle tentative de connexion est effectuée jusqu'à ce que le gestionnaire de files d'attente soit à nouveau en cours d'exécution.
2. Si le gestionnaire de files d'attente est configuré en tant que gestionnaire de files d'attente multi-instance, le client se reconnecte à l'instance active.

En se reconnectant automatiquement au même gestionnaire de files d'attente, la plupart des informations d'état que le gestionnaire de files d'attente détenait pour le compte du client, telles que les files d'attente qu'il avait ouvertes et la rubrique à laquelle il était abonné, sont restaurées. Si le client a ouvert une file d'attente de réponse dynamique pour recevoir une réponse à une demande, la connexion à la file d'attente de réponse est également restaurée.

Haute disponibilité de RDQM

Le gestionnaire de files d'attente de données répliquées (RDQM) est une solution à haute disponibilité disponible sur les plateformes Red Hat Enterprise Linux for x86-64 .

Une configuration de gestionnaire de files d'attente de données répliquées comprend trois serveurs configurés dans un groupe à haute disponibilité (HA), chacun contenant une instance du gestionnaire de files d'attente. Une instance est le gestionnaire de files d'attente en cours d'exécution, qui réplique de manière synchrone ses données sur les deux autres instances. Si le serveur qui exécute ce gestionnaire de files d'attente échoue, une autre instance du gestionnaire de files d'attente démarre et peut utiliser les données actuelles. Les trois instances du gestionnaire de files d'attente peuvent éventuellement partager une adresse IP flottante, de sorte que les clients n'ont besoin d'être configurés qu'avec une seule adresse

IP. Une seule instance du gestionnaire de files d'attente peut être exécutée à la fois, même si le groupe HA est partitionné en raison de problèmes réseau. Le serveur qui exécute le gestionnaire de files d'attente est le serveur 'principal', chacun des deux autres serveurs est un serveur 'secondaire'.

Trois noeuds sont utilisés pour réduire considérablement la possibilité d'une situation de déconnexion cérébrale. Dans un système à haute disponibilité à deux noeuds, une déconnexion cérébrale peut se produire lorsque la connectivité entre les deux noeuds est interrompue. Sans connectivité, les deux noeuds peuvent exécuter le gestionnaire de files d'attente en même temps, en accumulant des données différentes. Lorsque la connexion est restaurée, il existe deux versions différentes des données (un "split-brain"), et une intervention manuelle est nécessaire pour décider quel ensemble de données conserver et lequel supprimer.

Le gestionnaire de files d'attente de données répliquées utilise un système à trois noeuds avec quorum pour éviter la situation de déconnexion cérébrale. Les noeuds qui peuvent communiquer avec au moins l'un des autres noeuds forment un quorum. Les gestionnaires de files d'attente ne peuvent s'exécuter que sur un noeud disposant d'un quorum. Le gestionnaire de files d'attente ne peut pas s'exécuter sur un noeud qui n'est pas connecté à au moins un autre noeud. Il ne peut donc jamais s'exécuter sur deux noeuds en même temps:

- En cas de défaillance d'un seul noeud, le gestionnaire de files d'attente peut s'exécuter sur l'un des deux autres noeuds. Si deux noeuds échouent, le gestionnaire de files d'attente ne peut pas s'exécuter sur le noeud restant car le noeud ne possède pas de quorum (le noeud restant ne peut pas déterminer si les deux autres noeuds ont échoué ou s'ils sont toujours en cours d'exécution et qu'ils ont perdu la connectivité).
- Si un seul noeud perd la connectivité, le gestionnaire de files d'attente ne peut pas s'exécuter sur ce noeud car le noeud ne possède pas de quorum. Le gestionnaire de files d'attente peut s'exécuter sur l'un des deux noeuds restants, qui possèdent un quorum. Si tous les noeuds perdent la connectivité, le gestionnaire de files d'attente ne peut pas s'exécuter sur l'un des noeuds car aucun d'entre eux ne possède de quorum.

Remarque : IBM MQ Console ne prend pas en charge les gestionnaires de files d'attente de données répliquées. Vous pouvez utiliser IBM MQ Explorer avec des gestionnaires de files d'attente de données répliquées, mais cela n'affiche pas d'informations spécifiques aux fonctions des gestionnaires de files d'attente de données répliquées.

La configuration de groupe des trois noeuds est gérée par Pacemaker. La réplication entre les trois noeuds est gérée par DRBD. (Voir <https://clusterlabs.org/pacemaker/> pour plus d'informations sur Pacemaker et <https://docs.linbit.com/docs/users-guide-9.0/> pour plus d'informations sur DRBD.)

Vous pouvez sauvegarder vos gestionnaires de files d'attente de données répliquées à l'aide du processus décrit dans «Sauvegarde des données de gestionnaire de files d'attente», à la page 738. L'arrêt du gestionnaire de files d'attente et sa sauvegarde n'ont aucun effet sur la surveillance de noeud effectuée par la configuration du gestionnaire de files d'attente de données répliquées.

La figure ci-après illustre un déploiement type avec un gestionnaire de files d'attente de données répliquées exécuté sur chacun des trois noeuds du groupe HA.

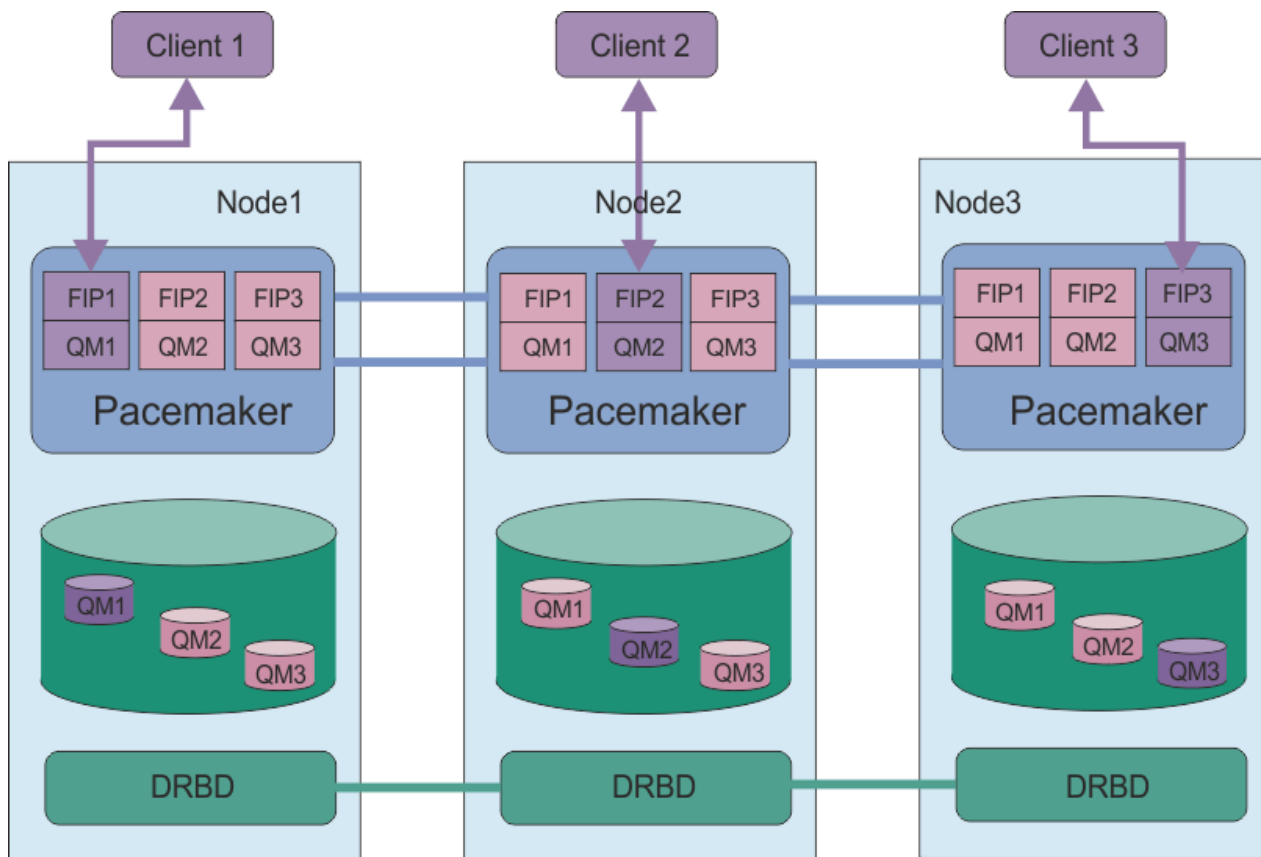


Figure 78. Exemple de groupe HA avec trois gestionnaires de files d'attente de données répliquées

Dans la figure suivante, Node3 a échoué, les connexions Pacemaker ont été perdues et le gestionnaire de files d'attente QM3 est exécuté sur Node2 à la place.

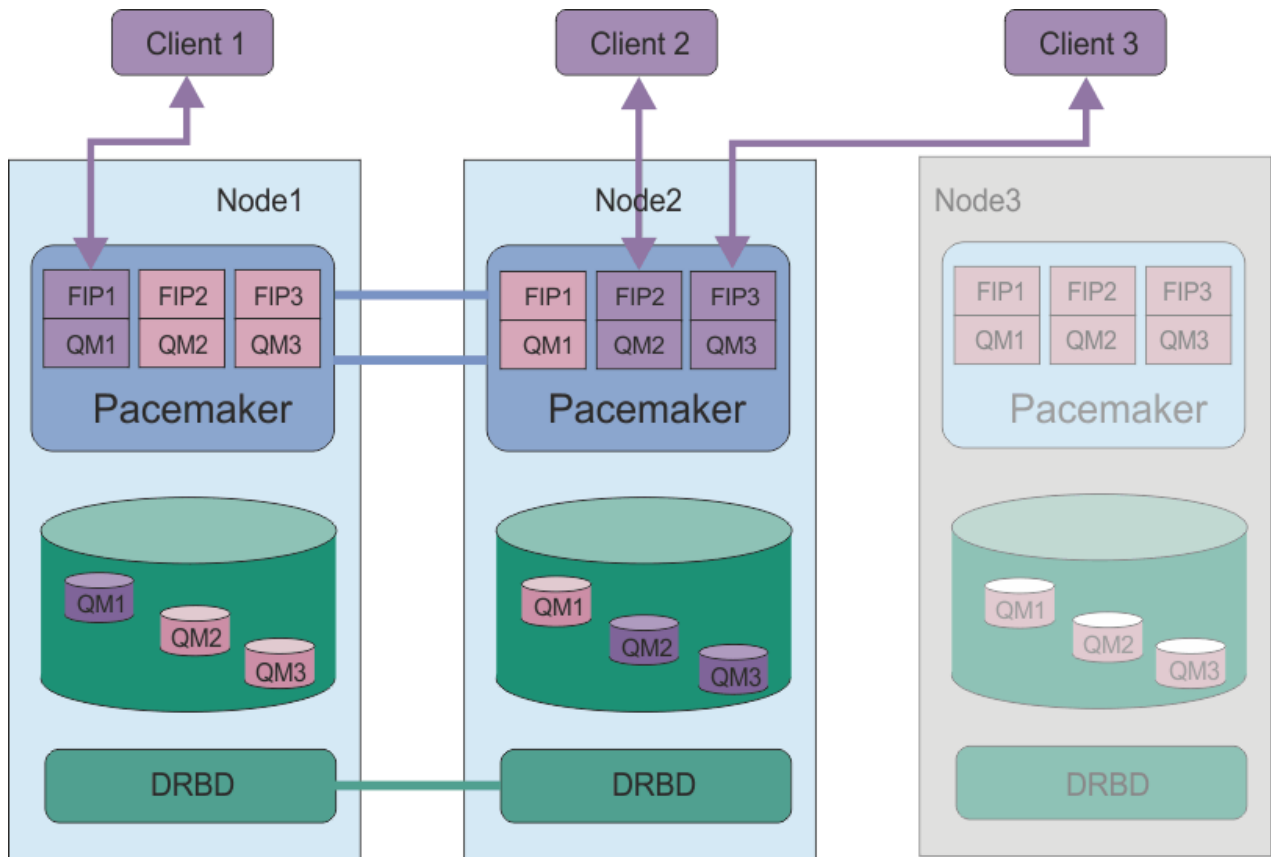


Figure 79. Exemple après échec de node3

Remarque : Lorsque les gestionnaires de files d'attente basculent sur un autre noeud, ils conservent l'état qu'ils avaient lors de la reprise en ligne. Les gestionnaires de files d'attente qui étaient en cours d'exécution sont démarrés et les gestionnaires de files d'attente qui ont été arrêtés restent arrêtés.

Tâches associées

[Installation de gestionnaires de files d'attente de données répliquées \(RDQM\)](#)

[Application de mises à jour de maintenance pour le gestionnaire de files d'attente de données répliquées \(RDQM\)](#)

[Migration des gestionnaires de files d'attente de données répliquées](#)

[Traitement des incidents liés aux configurations RDQM](#)

Linux **Conditions requises pour la solution RDQM HA**

Vous devez répondre à un certain nombre d'exigences avant de configurer le groupe à haute disponibilité des gestionnaires de files d'attente de données répliquées.

Configuration système requise

Avant de configurer le groupe à haute disponibilité des gestionnaires de files d'attente de données répliquées, vous devez effectuer une configuration sur chacun des trois serveurs qui doivent faire partie du groupe à haute disponibilité.

- Chaque noeud requiert un groupe de volumes nommé `drbdpool1`. Le stockage de chaque gestionnaire de files d'attente de données répliquées est alloué en tant que volume logique distinct par gestionnaire de files d'attente à partir de ce groupe de volumes. Pour des performances optimales, ce groupe de volumes doit être constitué d'un ou de plusieurs volumes physiques correspondant à des unités de disque internes (de préférence des unités SSD). Vous pouvez créer `drbdpool1` avant ou après avoir installé la solution à haute disponibilité des gestionnaires de files d'attente de données répliquées, mais vous devez créer `drbdpool1` avant de créer des gestionnaires de files d'attente de données répliquées.

Vérifiez la configuration de votre groupe de volumes à l'aide de la commande **vgs** . La sortie doit être similaire à la suivante :

```
VG          #PV #LV #SN Attr   VSize  VFree
drbdpool1  1   9   0 wz--n- <16.00g <7.00g
rhe1       1   2   0 wz--n- <15.00g  0
```

En particulier, vérifiez qu'il n'y a pas de caractère c dans la sixième colonne des attributs (c'est-à-dire, wz--nc). c indique que la mise en cluster est activée, et si c'est le cas, vous devez supprimer le groupe de volumes et le recréer sans la mise en cluster.

- Une fois que vous avez créé le groupe de volumes drbdpool1 , n'y faites rien d'autre. IBM MQ gère les volumes logiques créés dans drbdpool1 , ainsi que leur mode et leur emplacement de montage.
- Chaque noeud requiert jusqu'à trois interfaces qui sont utilisées pour configurer la prise en charge des gestionnaires de files d'attente de données répliquées:
 - Une interface principale pour Pacemaker pour surveiller le groupe HA.
 - Une autre interface permettant à Pacemaker de surveiller le groupe HA.
 - Une interface pour la réplication de données synchrone, appelée interface de réplication. Cette bande passante doit être suffisante pour prendre en charge les exigences de réplication compte tenu de la charge de travail attendue de tous les gestionnaires de files d'attente de données répliquées s'exécutant dans le groupe à haute disponibilité.

Vous pouvez configurer le groupe à haute disponibilité de sorte que la même adresse IP soit utilisée pour les trois interfaces, qu'une adresse IP distincte soit utilisée pour chaque interface ou que la même adresse IP soit utilisée pour l'interface principale et l'autre et qu'une adresse IP distincte soit utilisée pour l'interface de réplication.

Pour une tolérance maximale aux pannes, ces interfaces doivent être des cartes d'interface réseau (NIC) indépendantes.

- DRBD requiert que chaque noeud du groupe HA ait un nom d'hôte Internet valide (valeur renvoyée par uname -n), tel que défini par RFC 952 modifié par RFC 1123.
- S'il existe un pare-feu entre les noeuds du groupe HA, le pare-feu doit autoriser le trafic entre les noeuds sur une plage de ports. Un exemple de script est fourni, /opt/mqm/samp/rdqm/firewalld/configure.sh , qui ouvre les ports nécessaires si vous exécutez le pare-feu standard dans RHEL. Vous devez exécuter le script en tant que root. Si vous utilisez un autre pare-feu, examinez les définitions de service /usr/lib/firewalld/services/rdqm* pour voir quels ports doivent être ouverts. Le script ajoute les règles de service firewalld permanentes suivantes pour DRBD, Pacemaker et IBM MQ:
 - MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-drbd.xml autorise les ports TCP 7000-7100.
 - MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-pacemaker.xml autorise les ports UDP 5404 à 5407
 - MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-mq.xml autorise le port TCP 1414 (vous devez éditer le script si vous avez besoin d'un port différent)
- Si le système utilise SELinux en mode d'application, vous devrez peut-être exécuter la commande suivante:

```
semanage permissive -a drbd_t
```

Si vous avez installé le package drbd-selinux , vous n'avez pas besoin d'exécuter **semanage**. Ce package doit être installé sur chaque noeud ou **semanage** doit être exécuté sur chaque noeud.

Configuration requise pour le réseau

Il est recommandé de localiser les trois noeuds du groupe à haute disponibilité des gestionnaires de files d'attente de données répliquées dans le même centre de données.

Si vous choisissez de localiser les noeuds dans différents centres de données, tenez compte des limitations suivantes:

- Les performances se dégradent rapidement si le temps d'attente entre les centres de données augmente. Bien qu' IBM prenne en charge un temps d'attente pouvant atteindre 5 ms, vous pouvez constater que les performances de votre application ne peuvent pas tolérer plus de 1 à 2 ms de temps d'attente.
- Les données envoyées via le lien de réplication ne font pas l'objet d'un chiffrement supplémentaire au-delà de celui qui pourrait être en place lors de l'utilisation de IBM MQ AMS.

Vous pouvez éventuellement configurer une adresse IP flottante pour permettre à un client d'utiliser la même adresse IP pour un gestionnaire de files d'attente de données répliquées (RDQM) quel que soit le noeud du groupe à haute disponibilité sur lequel il s'exécute. L'adresse flottante est liée à une interface physique nommée sur le noeud principal du gestionnaire de files d'attente de données répliquées. Si le gestionnaire de files d'attente de données répliquées bascule et qu'un noeud différent devient le noeud principal, l'adresse IP flottante est liée à une interface du même nom sur le nouveau noeud principal. Les interfaces physiques sur les trois noeuds doivent toutes avoir le même nom et appartenir au même sous-réseau que l'adresse IP flottante.

Exigences utilisateur pour la configuration du cluster

Vous pouvez configurer le groupe HA des gestionnaires de files d'attente de données répliquées en tant qu'utilisateur `root`. Si vous ne souhaitez pas configurer en tant que `root`, configurez en tant qu'utilisateur dans le groupe `mqm` à la place. Pour qu'un utilisateur du groupe `mqm` puisse configurer le cluster de gestionnaires de files d'attente de données répliquées, vous devez respecter les exigences suivantes:

- L'utilisateur `mqm` doit pouvoir utiliser `sudo` pour exécuter des commandes sur chacun des trois serveurs qui constituent le groupe RDQM HA.
- Si l'utilisateur `mqm` peut utiliser SSH sans mot de passe pour exécuter des commandes sur chacun des trois serveurs qui constituent le groupe RDQM HA, l'utilisateur n'a besoin d'exécuter des commandes que sur l'un des serveurs.
- L'utilisateur `mqm` doit avoir le même ID utilisateur sur les trois serveurs.
- Le groupe `mqm` doit avoir le même GID sur les trois serveurs.

Vous devez configurer `sudo` pour que l'utilisateur `mqm` puisse exécuter les commandes suivantes avec les droits d'accès `root`:

```
/opt/mqm/bin/crtmqm
/opt/mqm/bin/dltmqm
/opt/mqm/bin/rdqmadm
/opt/mqm/bin/rdqmstatus
```

Exigences utilisateur pour l'utilisation des gestionnaires de files d'attente

Pour créer, supprimer ou configurer des gestionnaires de files d'attente de données répliquées (RDQM), vous devez utiliser un ID utilisateur appartenant aux groupes `mqm` et `haclient` (le groupe `haclient` est créé lors de l'installation de Pacemaker).

Linux

Configuration de l'accès SSH et sudo sans mot de passe

Vous pouvez configurer l'accès SSH et sudo sans mot de passe de sorte que vous n'ayez besoin d'émettre des commandes de configuration que sur un seul noeud du groupe à haute disponibilité. (La configuration d'un tel accès est facultative. Vous pouvez également exécuter des commandes sur chaque noeud.)

Pourquoi et quand exécuter cette tâche

Pour configurer une connexion SSH sans mot de passe, vous devez configurer l'ID `mqm` sur chaque noeud, puis générer une clé sur chaque noeud pour cet utilisateur. Vous distribuez ensuite les clés aux autres noeuds et vous testez la connexion pour ajouter chaque noeud à la liste des hôtes connus. Enfin, vous verrouillez l'ID `mqm`.

Remarque : Les instructions supposent que vous définissez un groupe à haute disponibilité avec des interfaces principale, secondaire et de réplication distinctes, et que vous définissez donc un accès SSH sans mot de passe sur les interfaces principale et secondaire. Si vous prévoyez de configurer un système avec une adresse IP unique, vous définissez un accès SSH sans mot de passe sur cette interface unique. Si vous prévoyez de configurer un système avec deux adresses IP pour HA_Primary et HA_Replication, le ssh doit être configuré pour l'adresse HA_Primary.

Vous pouvez ensuite créer un accès sudo pour l'ID mqm sur chaque noeud.

Procédure

1. Pour configurer SSH sans mot de passe:

a) Sur chacun des trois noeuds, procédez comme suit pour configurer l'utilisateur mqm et générer une clé SSH:

i) Remplacez le répertoire de base mqm par /home/mqm:

```
usermod -d /home/mqm mqm
```

ii) Créez le répertoire /home/mqm :

```
mkdir -p /home/mqm
```

iii) Ajoutez le mot de passe mqm :

```
passwd mqm
```

iv) Exécutez le shell interactif en tant que mqm:

```
su mqm
```

v) Générez la clé d'authentification mqm :

```
ssh-keygen -t rsa -f /home/mqm/.ssh/id_rsa -N ''
```

b) Sur chacun des trois noeuds, procédez comme suit pour ajouter la clé de ce noeud aux deux autres noeuds et tester les connexions pour chaque noeud principal et (si utilisé) les autres adresses:

i) Ajoutez la clé aux noeuds distants:

```
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node1_primary_address
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node1_alternate_address
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node2_primary_address
ssh-copy-id -i /home/mqm/.ssh/id_rsa.pub remote_node2_alternate_address
```

ii) Vérifiez le ssh sans mot de passe et mettez à jour known_hosts pour les noeuds distants:

```
ssh remote_node1_primary_address uname -n
ssh remote_node1_alternate_address uname -n
ssh remote_node2_primary_address uname -n
ssh remote_node2_alternate_address uname -n
```

Pour chaque connexion, vous êtes invité à confirmer que vous souhaitez continuer. Confirmez la mise à jour des hôtes connus pour chacun d'eux. Vous devez effectuer cette opération avant de tenter de configurer le groupe HA à l'aide de SSH sans mot de passe.

iii) Quittez le shell interactif en tant que mqm:

```
exit
```

c) Sur chaque noeud, en tant que superutilisateur, procédez comme suit pour supprimer le mot de passe mqm et verrouiller l'ID:

i) Supprimez le mot de passe mqm :

```
passwd -d mqm
```

ii) Verrouiller mqm:

```
passwd -l mqm
```

2. Sur chaque noeud, en tant que superutilisateur, configurez l'accès sudo pour l'utilisateur mqm en créant le fichier `/etc/sudoers.d/mqm` contenant le texte suivant:

```
mqm ALL=(root) NOPASSWD: /opt/mqm/bin/crtmqm, /opt/mqm/bin/dltmqm, /opt/mqm/bin/rdqmadm, /opt/mqm/bin/rdqmstatus, /opt/mqm/bin/rdqmdi
```

Linux Définition du cluster Pacemaker (groupe HA)

Le groupe HA est un cluster Pacemaker . Vous définissez le cluster Pacemaker en éditant le fichier `/var/mqm/rdqm.ini` et en exécutant la commande `rdqmadm` .

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur Pacemaker, voir <https://clusterlabs.org/pacemaker/> . Vous pouvez créer le cluster Pacemaker en tant qu'utilisateur du groupe mqm si l'utilisateur mqm peut utiliser sudo. Si l'utilisateur peut également établir une connexion SSH à chaque serveur sans mot de passe, il suffit de modifier le fichier `rdqm.ini` et d'exécuter `rdqmadm` sur l'un des serveurs pour créer le cluster Pacemaker . Sinon, vous devez créer le fichier et exécuter la commande en tant que `root` sur chacun des serveurs qui doivent être des noeuds.

Le fichier `rdqm.ini` indique les adresses IP utilisées par le gestionnaire de files d'attente de données répliquées pour les noeuds du cluster Pacemaker . Vous devez indiquer le nom de chaque noeud, qui doit être le nom d'hôte renvoyé par la commande `uname -n` .

Un groupe à haute disponibilité de gestionnaire de files d'attente de données répliquées peut être configuré pour utiliser une, deux ou trois adresses IP:

- Une adresse IP: les signaux de présence et la réplication partagent le même lien
- Deux adresses IP: les signaux de présence et la réplication utilisent des liens distincts
- Trois adresses IP: un lien pour la réplication et deux liens distincts pour les signaux de présence

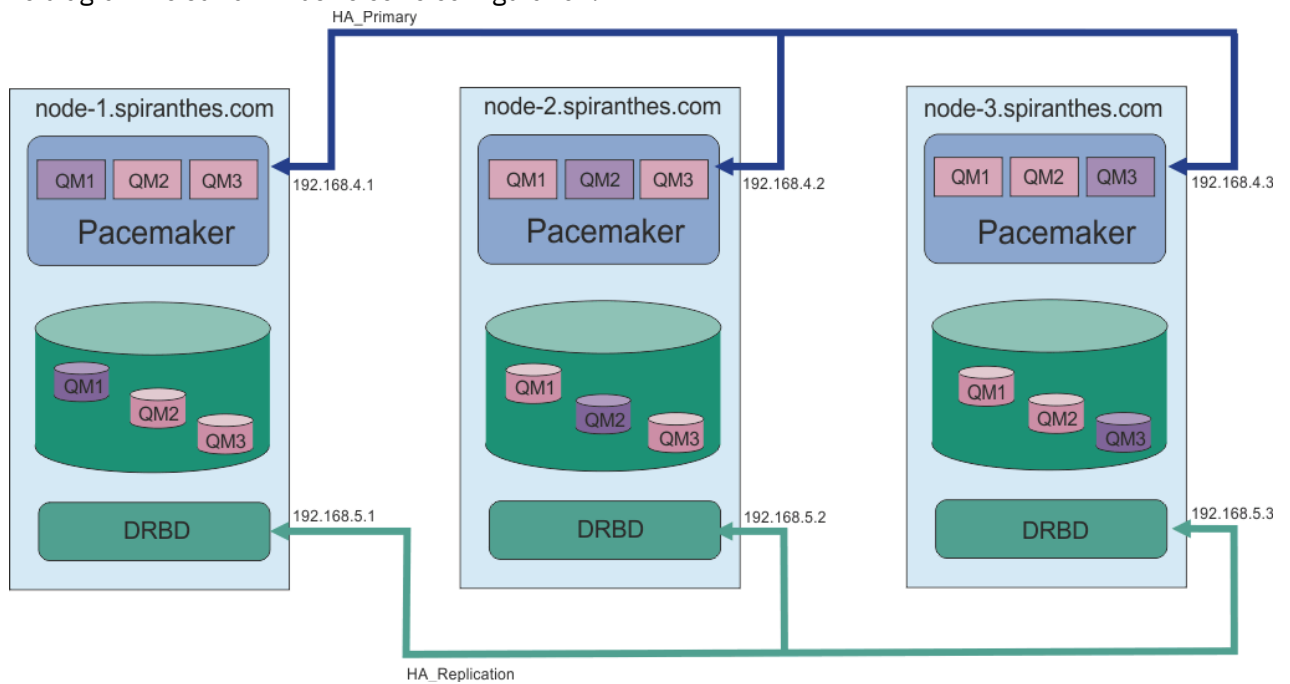
Ces options sont fournies pour prendre en charge différents modèles de déploiement pour les gestionnaires de files d'attente de données répliquées. Les différentes options peuvent être utilisées pour optimiser la résilience de la solution RDQM en fonction de l'environnement utilisé. Les configurations qui utilisent deux ou trois adresses IP sont principalement destinées aux déploiements pour lesquels un contrôle granulaire est requis sur lesquels le réseau physique relie les signaux de présence et le trafic de réplication est utilisé pour configurer la redondance de la connectivité entre les noeuds. Une connectivité à haute disponibilité et résiliente peut également être implémentée au niveau de la couche réseau, par exemple en utilisant l'agrégation de liaisons. Avec l'agrégation de liaisons, plusieurs liaisons réseau physiques sont utilisées pour fournir une liaison logique unique qui peut continuer à fonctionner en cas d'échec de liaisons physiques individuelles. Si le gestionnaire de files d'attente de données répliquées est déployé dans un environnement où la connectivité du réseau est virtualisée et / ou où la connectivité résiliente est implémentée au niveau de la couche réseau, il est généralement préférable d'utiliser une adresse IP unique pour les signaux de présence et la réplication.

L'exemple suivant illustre l'utilisation de deux adresses IP. Votre fichier `rdqm.ini` comporte une zone `HA_Primary` et une zone `HA_Replication` pour chaque noeud, mais pas de zone `HA_Alternate` :

```
Node:
  Name=rdqm-node-1.spiranthes.com
  HA_Primary=192.168.4.1
  HA_Replication=192.168.5.1
Node:
  Name=rdqm-node-2.spiranthes.com
  HA_Primary=192.168.4.2
  HA_Replication=192.168.5.2
Node:
  Name=rdqm-node-3.spiranthes.com
  HA_Primary=192.168.4.3
```

HA_Replication=192.168.5.3

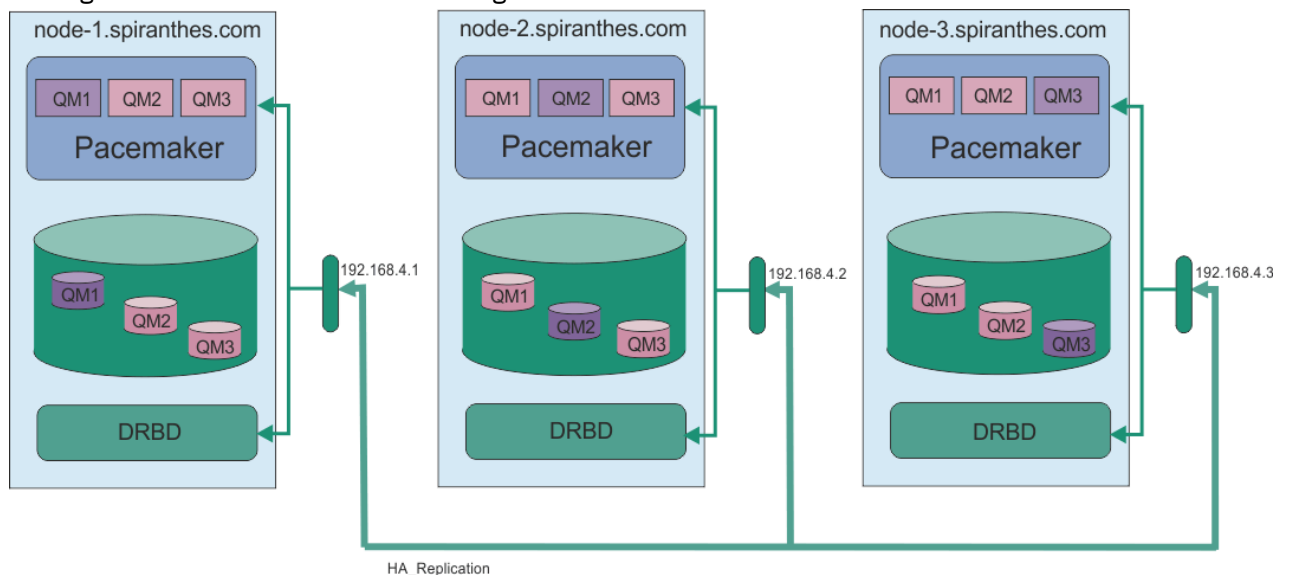
Le diagramme suivant illustre cette configuration:



L'exemple de fichier suivant illustre la configuration d'un exemple de cluster Pacemaker qui utilise l'interface HA_Replication pour la surveillance. Dans ce cas, vous ne spécifiez que l'interface HA_Replication :

```
Node:  
  Name=rdqm-node-1.spiranthes.com  
  HA_Replication=192.168.4.1  
Node:  
  Name=rdqm-node-2.spiranthes.com  
  HA_Replication=192.168.4.2  
Node:  
  Name=rdqm-node-3.spiranthes.com  
  HA_Replication=192.168.4.3
```

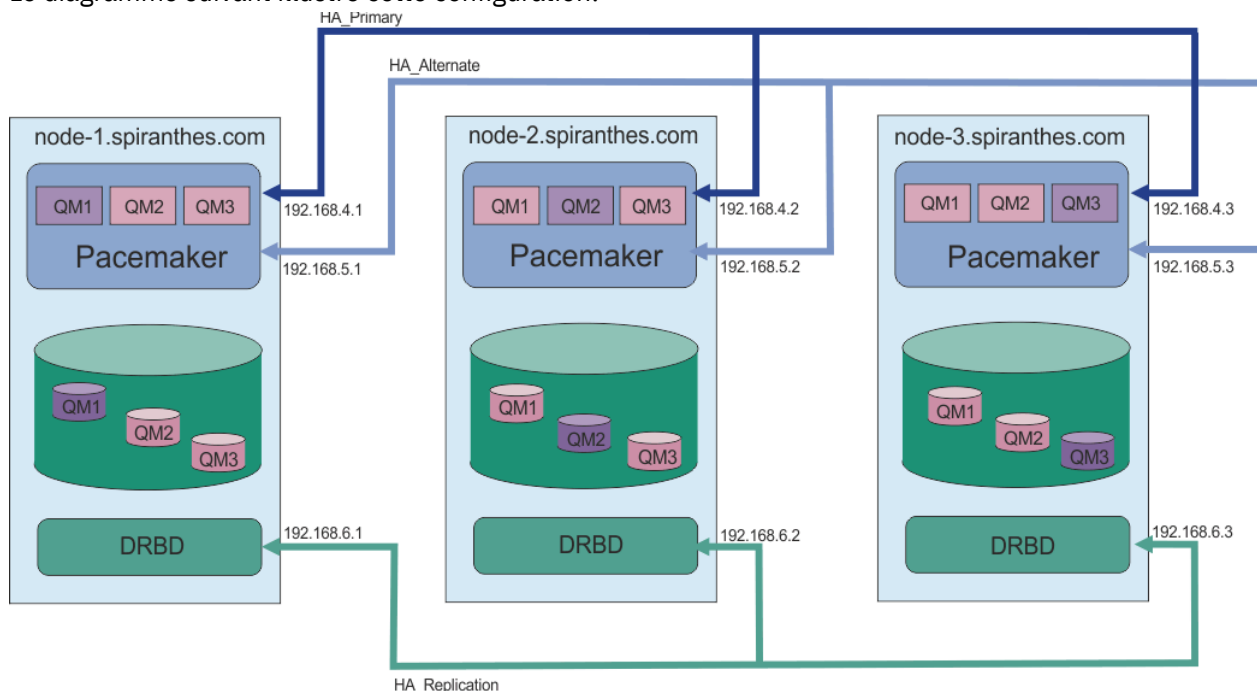
Le diagramme suivant illustre cette configuration:



L'exemple de fichier suivant illustre la configuration d'un exemple de cluster Pacemaker qui utilise une adresse IP distincte pour chaque interface:

```
Node:
Name=rdqm-node-1.spiranthes.com
HA_Primary=192.168.4.1
HA_Alternate=192.168.5.1
HA_Replication=192.168.6.1
Node:
Name=rdqm-node-2.spiranthes.com
HA_Primary=192.168.4.2
HA_Alternate=192.168.5.2
HA_Replication=192.168.6.2
Node:
Name=rdqm-node-3.spiranthes.com
HA_Primary=192.168.4.3
HA_Alternate=192.168.5.3
HA_Replication=192.168.6.3
```

Le diagramme suivant illustre cette configuration:



L'ordre dans lequel vous spécifiez les noeuds doit être le même dans tous les fichiers `rdqm.ini` de votre configuration. Vos trois noeuds doivent avoir une vue commune de celui qui est Node1, celui qui est Node2, etc.

Procédure

- Pour définir le cluster Pacemaker en tant qu'utilisateur `root`:
 - a) Editez le fichier `/var/mqm/rdqm.ini` sur l'un des trois serveurs afin que le fichier définisse le cluster.
 - b) Copiez le fichier sur les deux autres serveurs qui seront des noeuds du cluster Pacemaker.
 - c) Exécutez la commande suivante en tant que `root` sur chacun des trois serveurs:

```
rdqmadm -c
```

- Pour définir le cluster Pacemaker en tant qu'utilisateur dans le groupe `mqm` sur chaque noeud:
 - a) Vérifiez que l'utilisateur `mqm` peut utiliser **sudo** pour exécuter des commandes.
 - b) Editez le fichier `/var/mqm/rdqm.ini` sur l'un des trois serveurs afin qu'il définisse le cluster Pacemaker.

- c) Copiez le fichier `/var/mqm/rdqm.ini` sur les deux autres serveurs qui seront des noeuds dans le cluster Pacemaker .
- d) Exécutez la commande suivante sur chaque serveur:

```
rdqmadm -c
```

- Pour définir le cluster Pacemaker en tant qu'utilisateur dans le groupe `mqm` à partir d'un noeud:
 - a) Vérifiez que l'utilisateur `mqm` peut utiliser **sudo** pour exécuter des commandes et qu'il peut se connecter à chaque serveur à l'aide de SSH sans mot de passe.
 - b) Editez le fichier `/var/mqm/rdqm.ini` sur l'un des trois serveurs afin qu'il définisse le cluster Pacemaker .
 - c) Exécutez ensuite la commande suivante :

```
rdqmadm -c
```

Référence associée

[rdqmadm \(administrer un cluster de gestionnaires de files d'attente de données répliquées\)](#)

Linux *Suppression du cluster Pacemaker (groupe HA)*

Le groupe HA est un cluster Pacemaker . Vous pouvez supprimer une configuration de cluster Pacemaker en exécutant la commande **rdqmadm** avec l'option `-u` .

Pourquoi et quand exécuter cette tâche

Vous ne pouvez pas supprimer la configuration de cluster Pacemaker si des gestionnaires de files d'attente de données répliquées existent toujours sur l'un des noeuds.

Procédure

- Pour supprimer la configuration de cluster Pacemaker , entrez la commande suivante à partir de l'un des noeuds:

```
rdqmadm -u
```

Référence associée

[rdqmadm \(administrer un cluster de gestionnaires de files d'attente de données répliquées\)](#)

Linux *Créer un gestionnaire de files d'attente de données répliquées*

La commande **crtmqm** permet de créer un gestionnaire de files d'attente de données répliquées à haute disponibilité (RDQM).

Pourquoi et quand exécuter cette tâche

Vous pouvez créer un gestionnaire de files d'attente de données répliquées à haute disponibilité (RDQM) en tant qu'utilisateur dans le groupe `mqm` si l'utilisateur `mqm` peut utiliser `sudo`. Si l'utilisateur peut également établir une connexion SSH à chaque noeud sans mot de passe, il suffit d'exécuter la commande de création de gestionnaire de files d'attente de données répliquées sur un noeud pour créer le gestionnaire de files d'attente de données répliquées sur les trois noeuds. Sinon, vous devez être `root` pour créer un gestionnaire de files d'attente de données répliquées et vous devez exécuter des commandes sur les trois noeuds.

Remarque : Il existe une limite absolue de 129 gestionnaires de files d'attente dans un groupe à haute disponibilité. Si vous tentez de créer plus que cela, la tentative échouera. Dans la pratique, l'ajout de plus de 50 gestionnaires de files d'attente à un groupe à haute disponibilité peut entraîner des problèmes de délai d'attente.

Les points suivants fournissent des conseils sur le dimensionnement du système de fichiers du gestionnaire de files d'attente:

1. Lors de la création d'un gestionnaire de files d'attente de données répliquées, un système de fichiers est alloué pour stocker les données et les journaux du gestionnaire de files d'attente. Il est important de dimensionner ce système de fichiers de manière appropriée afin que le gestionnaire de files d'attente puisse enregistrer l'activité en cours dans ses journaux et stocker les messages d'application dans les files d'attente. Lors de la définition de la taille du système de fichiers, tenez compte des pics de besoins en matière de messagerie, de la croissance future de la charge de travail et des indisponibilités d'application susceptibles de générer des messages dans les files d'attente. Pour savoir comment calculer la taille du journal de reprise du gestionnaire de files d'attente, voir [«Quelle est la taille de mon système de fichiers journaux?»](#), à la page 715. Lors du calcul des exigences de stockage pour les messages d'application, la taille et le nombre de messages doivent être pris en compte, ainsi que leur en-tête MQMD et les propriétés de message qu'ils possèdent.
2. Les systèmes de fichiers du gestionnaire de files d'attente de données répliquées ne peuvent pas être redimensionnés dynamiquement. Vous devez sauvegarder puis restaurer un gestionnaire de files d'attente de données répliquées (RDQM) avec un système de fichiers plus grand si nécessaire. Voir [«Redimensionnement du système de fichiers pour un gestionnaire de files d'attente de données répliquées à haute disponibilité»](#), à la page 633.
3. Vous pouvez limiter la taille des files d'attente individuelles sur le disque en utilisant des attributs de file d'attente locale, tels que MAXDEPTH et MAXFSIZE. Voir [Modification des fichiers de file d'attente IBM MQ](#).
4. Vous devez surveiller votre utilisation continue du disque et répondre de manière appropriée si l'utilisation du disque augmente avant que l'utilisation du système de fichiers ne devienne critique. L'utilisation du système de fichiers peut être surveillée à l'aide des fonctions de plateforme / système d'exploitation ou en s'abonnant aux métriques publiées dans les rubriques système IBM MQ décrites dans [Métriques publiées sur les rubriques système](#).

Procédure

- Pour créer un gestionnaire de files d'attente de données répliquées en tant qu'utilisateur dans le groupe mqm :
 - a) Vérifiez que l'utilisateur mqm peut utiliser **sudo** pour exécuter des commandes et peut se connecter à chaque serveur à l'aide de SSH sans mot de passe.
 - b) Entrez la commande suivante :

```
crtmqm -sx [-fs FilesystemSize] qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées. Vous pouvez éventuellement spécifier la taille du système de fichiers pour le gestionnaire de files d'attente (c'est-à-dire la taille du volume logique créé dans le groupe de volumes drbdpool).

La commande tente d'utiliser SSH pour se connecter aux autres noeuds du cluster en tant qu'utilisateur mqm . Si la connexion aboutit, les instances secondaires du gestionnaire de files d'attente sont créées sur les noeuds. Sinon, vous devez créer les instances secondaires, puis exécuter la commande **crtmqm -sx** (comme décrit pour l'utilisateur root).

- Pour créer un gestionnaire de files d'attente de données répliquées en tant qu'utilisateur root:
 - a) Entrez la commande suivante sur chacun des noeuds qui doivent héberger les instances secondaires du gestionnaire de files d'attente de données répliquées:

```
crtmqm -sxs [-fs FilesystemSize] qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées. Vous pouvez éventuellement spécifier la taille du système de fichiers pour le gestionnaire de files d'attente (c'est-à-dire la taille du volume logique créé dans le groupe de volumes drbdpool). Vous devez spécifier la même taille de système de fichiers pour le gestionnaire de files d'attente de données répliquées (RDQM) sur les trois noeuds du groupe HA. La taille est une valeur numérique, spécifiée en Go. Vous pouvez spécifier une valeur en Mo en entrant la valeur suivie du caractère M

La commande crée une instance secondaire du gestionnaire de files d'attente de données répliquées.

b) Sur le noeud restant, entrez la commande suivante:

```
crtmqm -sx [-fs FilesystemSize] qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées. Vous pouvez éventuellement spécifier la taille du système de fichiers pour le gestionnaire de files d'attente. La taille est une valeur numérique, spécifiée en Go. Vous pouvez indiquer une valeur en Mo en entrant la valeur suivie du caractère M.

La commande détermine si l'instance secondaire du gestionnaire de files d'attente existe sur les deux autres noeuds. S'il existe des secondaires, la commande crée et démarre le gestionnaire de files d'attente principal. Si les noeuds secondaires n'existent pas, vous êtes invité à exécuter la commande **crtmqm -sxs** sur chacun des noeuds.

En dehors des arguments DataPath (**-md**) et LogPath (**-ld**), tous les arguments valides pour la création d'un gestionnaire de files d'attente Linux standard sont également valides pour un gestionnaire de files d'attente de données répliquées principal.

Remarque : Lorsque vous créez un gestionnaire de files d'attente de données répliquées, le prochain numéro de port disponible au-dessus de 7000 est alloué pour le lien de réplication. S'il est découvert que le port choisi est utilisé par une autre application, la commande **crtmqm** échoue avec l'erreur AMQ6543 et ce port est ajouté à une liste d'exclusion. Vous devez supprimer les instances secondaires du gestionnaire de files d'attente, puis exécuter à nouveau la commande **crtmqm**.

Référence associée

[crtmqm](#)

Linux *Suppression d'un RDQM à haute disponibilité*

La commande **dltmqm** permet de supprimer un gestionnaire de files d'attente de données répliquées à haute disponibilité (RDQM).

Pourquoi et quand exécuter cette tâche

Vous devez exécuter la commande pour supprimer le gestionnaire de files d'attente de données répliquées sur le noeud principal du gestionnaire de files d'attente de données répliquées. Le gestionnaire de files d'attente de données répliquées doit d'abord être arrêté. Vous pouvez exécuter la commande en tant qu'utilisateur mqm si cet utilisateur dispose des privilèges sudo nécessaires. Sinon, vous devez exécuter la commande en tant que root. Une fois que les ressources associées au gestionnaire de files d'attente principal ont été supprimées, la commande tente de supprimer les gestionnaires de files d'attente secondaires à l'aide de ssh pour se connecter aux autres noeuds. Si cette suppression échoue, vous devez exécuter **dltmqm** manuellement sur les autres noeuds pour terminer le processus. Sur un noeud secondaire, la commande échoue si le gestionnaire de files d'attente principal n'a pas déjà été supprimé.

Procédure

- Pour supprimer un gestionnaire de files d'attente de données répliquées, entrez la commande suivante:

```
dltmqm RDQM_name
```

Référence associée

[dltmqm](#)

Vous pouvez migrer un gestionnaire de files d'attente existant pour qu'il devienne un gestionnaire de files d'attente de données répliquées (RDQM) à haute disponibilité en sauvegardant ses données persistantes, puis en restaurant les données dans un gestionnaire de files d'attente de données répliquées nouvellement créé portant le même nom.

Pourquoi et quand exécuter cette tâche

Les gestionnaires de files d'attente de données répliquées à haute disponibilité requièrent un volume logique dédié (système de fichiers) et la configuration de la réplication de disque et du contrôle à haute disponibilité. Ces composants sont configurés uniquement lorsqu'un nouveau gestionnaire de files d'attente est créé. Un gestionnaire de files d'attente existant peut être migré pour utiliser un gestionnaire de files d'attente de données répliquées en sauvegardant ses données persistantes, puis en restaurant les données dans un gestionnaire de files d'attente de données répliquées nouvellement créé et portant le même nom. Cette procédure permet de conserver la configuration, l'état et les messages persistants du gestionnaire de files d'attente au moment de la création de la sauvegarde.

Remarque : Vous ne pouvez migrer un gestionnaire de files d'attente qu'à partir d'une version d' IBM MQ identique ou antérieure à la version sur laquelle le gestionnaire de files d'attente de données répliquées est installé. Le système d'exploitation et l'architecture doivent également être identiques. Sinon, vous devez créer un gestionnaire de files d'attente sur votre plateforme cible. Voir [Déplacement d'un gestionnaire de files d'attente vers un autre système d'exploitation](#).

Vous devez remplir les conditions suivantes avant de migrer un gestionnaire de files d'attente:

- Évaluez vos exigences de haute disponibilité et voir [«Haute disponibilité de RDQM»](#), à la page 616.
- Passez en revue les applications et les gestionnaires de files d'attente qui se connectent au gestionnaire de files d'attente. Prenez en compte les modifications requises pour router les connexions vers le noeud RDQM sur lequel le gestionnaire de files d'attente est en cours d'exécution. Par exemple, si vous configurez la haute disponibilité des gestionnaires de files d'attente de données répliquées (RDQM), vous pouvez envisager d'utiliser une adresse IP flottante. Voir [«Création et suppression d'une adresse IP flottante»](#), à la page 637.
- Mettez à disposition ou identifiez des noeuds de gestionnaire de files d'attente de données répliquées existants pour la configuration choisie. Pour plus d'informations sur la configuration système requise pour les gestionnaires de files d'attente de données répliquées, voir [«Conditions requises pour la solution RDQM HA»](#), à la page 619.
- Installez IBM MQ Advanced, qui inclut la fonction RDQM, sur chaque noeud.
- Configurez la configuration du groupe HA des gestionnaires de files d'attente de données répliquées (voir [«Définition du cluster Pacemaker \(groupe HA\)»](#), à la page 623).
- Vous pouvez éventuellement vérifier la configuration du gestionnaire de files d'attente de données répliquées à l'aide d'un gestionnaire de files d'attente de test, qui peut ensuite être supprimé. Il est recommandé de tester la configuration afin d'identifier et de résoudre les problèmes avant de migrer le gestionnaire de files d'attente.
- Vérifiez la configuration de sécurité du gestionnaire de files d'attente, puis répliquez les utilisateurs et les groupes locaux requis sur chaque noeud de gestionnaire de files d'attente de données répliquées.
- Vérifiez la configuration du gestionnaire de files d'attente et du canal pour déterminer si des exits d'API, des exits de canal ou des exits de conversion de données sont utilisés. Installez les exits requis sur chaque noeud RDQM.
- Passez en revue les services de gestionnaire de files d'attente qui ont été définis, puis installez et configurez les processus requis sur chaque noeud RDQM.

Procédure

1. Sauvegardez le gestionnaire de files d'attente existant:

- a) Arrêtez le gestionnaire de files d'attente existant en émettant une commande d'arrêt d'attente `endmqm -wou` une commande d'arrêt immédiat `endmqm -i`. Cette étape est importante pour garantir la cohérence des données de la sauvegarde.
- b) Déterminez l'emplacement du répertoire de données du gestionnaire de files d'attente en affichant le fichier de configuration IBM MQ , `mqs.ini`. Sous Linux, ce fichier se trouve dans le répertoire `/var/mqm`. Pour plus d'informations sur `mqs.ini`, voir [«Fichier de configuration IBM MQ , mqs.ini»](#), à la page 98.

Recherchez la section `QueueManager` du gestionnaire de files d'attente dans le fichier. Si la strophe contient une clé nommée `DataPath`, sa valeur est le répertoire de données du gestionnaire de files d'attente. Si la clé n'existe pas, le répertoire de données du gestionnaire de files d'attente peut être déterminé à l'aide des valeurs des clés `Prefix` et `Directory`. Le répertoire de données du gestionnaire de files d'attente est une concaténation de ces valeurs, au format `préfixe/qmgrs/répertoire`. Pour plus d'informations sur la section `QueueManager`, voir [«Section QueueManager du fichier mqs.ini»](#), à la page 109.

- c) Créez une sauvegarde du répertoire de données du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande `tar`. Par exemple, pour sauvegarder le répertoire de données d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (point) unique:

```
tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
```

- d) Déterminez l'emplacement du répertoire de journaux du gestionnaire de files d'attente en affichant le IBM MQ fichier de configuration du gestionnaire de files d'attente `qm.ini`. Ce fichier se trouve dans le répertoire de données du gestionnaire de files d'attente. Pour plus d'informations sur le fichier, voir [«Fichiers de configuration du gestionnaire de files d'attente, qm.ini»](#), à la page 111.

Le répertoire de journaux du gestionnaire de files d'attente est défini en tant que valeur de la clé `LogPath` dans la section `Log`. Pour plus d'informations sur la strophe, voir [«Section de journal du fichier qm.ini»](#), à la page 147.

- e) Créez une sauvegarde du répertoire de journaux du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande `tar`. Par exemple, pour sauvegarder le répertoire de journaux d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (point) unique:

```
tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
```

- f) Créez une sauvegarde des référentiels de certificats utilisés par le gestionnaire de files d'attente s'ils ne se trouvent pas dans le répertoire de données du gestionnaire de files d'attente. Assurez-vous que le fichier de la base de données de clés et le fichier de mot de passe secret sont sauvegardés. Pour plus d'informations sur le référentiel de clés du gestionnaire de files d'attente, voir [Référentiel de clés SSL/TLS](#) et [Localisation du référentiel de clés pour un gestionnaire de files d'attente](#). Pour plus d'informations sur la localisation du magasin de clés AMS si le gestionnaire de files d'attente est configuré pour utiliser l'interception MCA (Message Channel Agent) AMS, voir [Interception MCA \(Message Channel Agent\)](#).

- g) Le gestionnaire de files d'attente existant n'est plus nécessaire et peut donc être supprimé. Toutefois, dans la mesure du possible, vous ne devez supprimer le gestionnaire de files d'attente existant qu'une fois qu'il a été correctement restauré sur le système cible. Le report de la suppression garantit que le gestionnaire de files d'attente peut être redémarré si le processus de migration n'aboutit pas.

Remarque : Si vous reportez la suppression du gestionnaire de files d'attente existant, ne le redémarrez pas. Il est important que le gestionnaire de files d'attente reste arrêté car d'autres modifications apportées à sa configuration ou à son état sont perdues lors de la migration.

2. Préparez le noeud RDQM principal:

- a) Créez un gestionnaire de files d'attente de données répliquées (RDQM) portant le même nom que le gestionnaire de files d'attente que vous avez sauvegardé. Vérifiez que le système de fichiers alloué au gestionnaire de files d'attente de données répliquées par `crtmqm` est suffisamment grand

pour contenir les données, les journaux principaux et les journaux secondaires du gestionnaire de files d'attente existant, ainsi qu'un espace supplémentaire pour une extension ultérieure. Pour plus d'informations sur la création d'un gestionnaire de files d'attente de données répliquées, voir [«Créer un gestionnaire de files d'attente de données répliquées»](#), à la page 626.

- b) Déterminez le noeud de gestionnaire de files d'attente de données répliquées principal pour le gestionnaire de files d'attente. Pour plus d'informations sur la détermination du noeud principal, voir `rdqmstatus (display RDQM status)`.
 - c) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, si le gestionnaire de files d'attente de données répliquées est démarré, arrêtez-le à l'aide de la commande `endmqm -w` ou `endmqm -i`.
 - d) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, déterminez l'emplacement des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées (utilisez les méthodes décrites dans les étapes 1b et 1d).
 - e) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, supprimez le contenu des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées, mais pas les répertoires eux-mêmes.
3. Restaurez le gestionnaire de files d'attente sur le noeud RDQM principal:

- a) Copiez les sauvegardes des données et des répertoires de journaux du gestionnaire de files d'attente sur le noeud RDQM principal, ainsi que les sauvegardes distinctes des référentiels de certificats utilisés par le gestionnaire de files d'attente.
- b) Restaurez la sauvegarde du répertoire de données du gestionnaire de files d'attente dans le répertoire de données vide du nouveau gestionnaire de files d'attente de données répliquées, en veillant à préserver la propriété et les droits d'accès aux fichiers. Si la sauvegarde a été créée à l'aide de l'exemple de commande `tar` à l'étape 1c, la commande suivante peut être utilisée par le superutilisateur pour la restaurer:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

- c) Restaurez la sauvegarde du répertoire de journaux du gestionnaire de files d'attente dans le répertoire de journaux vide du nouveau gestionnaire de files d'attente de données répliquées, en veillant à préserver la propriété et les droits d'accès aux fichiers. Si la sauvegarde a été créée à l'aide de l'exemple de commande `tar` de l'étape 1e, la commande suivante peut être utilisée par le superutilisateur pour la restaurer:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

- d) Editez le fichier de configuration de gestionnaire de files d'attente restauré, `qm.ini`, dans le répertoire de données du gestionnaire de files d'attente de données répliquées. Mettez à jour la valeur de la clé `LogPath` dans la section `Log` pour spécifier le répertoire de journaux du gestionnaire de files d'attente de données répliquées (RDQM).

Passez en revue les autres chemins de fichier définis dans le fichier de configuration et mettez-les à jour si nécessaire. Par exemple, vous devrez peut-être mettre à jour les chemins suivants:

- Chemin d'accès aux fichiers journaux d'erreurs générés par les services de message de diagnostic.
 - Chemin d'accès aux exits requis par le gestionnaire de files d'attente.
 - Chemin d'accès aux fichiers de commutation de chargement si le gestionnaire de files d'attente est un coordinateur de transaction XA.
- e) Si le gestionnaire de files d'attente est configuré pour utiliser l'interception AMS Message Channel Agent (MCA), copiez le magasin de clés AMS dans la nouvelle installation du gestionnaire de files d'attente de données répliquées, puis vérifiez et mettez à jour la configuration. Le magasin de clés doit être disponible sur chaque noeud de gestionnaire de files d'attente de données répliquées. Par conséquent, s'il ne se trouve pas dans le système de fichiers répliqué pour le gestionnaire de files d'attente, il doit être copié sur chaque noeud. Pour plus d'informations, voir [Interception de l'agent MCA \(Message Channel Agent\)](#).

- f) Vérifiez que le gestionnaire de files d'attente est affiché par la commande **dspmq** et que son statut est signalé comme étant arrêté. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente à haute disponibilité de gestionnaire de files d'attente de données répliquées:

```
$ dspmq -o status -o ha
QMNAME(QM1) STATUS(Ended normally) HA(Replicated)
```

- g) Vérifiez que les données du gestionnaire de files d'attente restaurées ont été répliquées sur les noeuds de gestionnaire de files d'attente de données répliquées secondaires à l'aide de la commande **rdqmstatus** pour afficher le statut du gestionnaire de files d'attente. Le statut de la haute disponibilité doit être signalé comme Normal sur chaque noeud. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente à haute disponibilité de gestionnaire de files d'attente de données répliquées:

```
$ rdqmstatus -m QM1
Node:                               mqhavam10-adm
Queue manager status:               Ended normally
Queue manager file system:          50MB used, 0.2GB allocated [42%]
HA role:                             Primary
HA status:                           Normal
HA control:                           Disabled
HA current location:                 This node
HA preferred location:                This node
HA floating IP interface:             None
HA floating IP address:               None

Node:                               mqhavam11-adm
HA status:                            Normal

Node:                               mqhavam12-adm
HA status:                            Normal
```

- h) Démarrez le gestionnaire de files d'attente sur le noeud RDQM principal.
- i) Connectez-vous au gestionnaire de files d'attente et mettez à jour la valeur de l'attribut de gestionnaire de files d'attente SSLKEYR pour spécifier le nouvel emplacement du référentiel de certificats du gestionnaire de files d'attente. Par défaut, la valeur de cet attribut est *queue_manager_data_directory/ssl/key*. Le référentiel de certificats doit se trouver au même emplacement sur chaque noeud RDQM. Si le référentiel ne se trouve pas dans le système de fichiers répliqué pour le gestionnaire de files d'attente, il doit être copié sur chaque noeud à la place.
- j) Passez en revue les définitions d'objet IBM MQ pour le gestionnaire de files d'attente et mettez à jour la valeur des attributs d'objet qui font référence aux paramètres réseau modifiés, au répertoire d'installation de IBM MQ ou au répertoire de données du gestionnaire de files d'attente, y compris les objets suivants:
- Adresses IP locales utilisées par les programmes d'écoute (attributIPADDR).
 - Adresses IP locales utilisées par les canaux (attributLOCLADDR).
 - Adresses IP locales définies pour les canaux récepteurs de cluster (attributCONNAME).
 - Adresses IP locales définies pour les objets d'informations de communication (attributGRPADDR).
 - Chemins système définis pour les définitions d'objet de processus et de service.
- k) Arrêtez et redémarrez le gestionnaire de files d'attente pour vous assurer que les modifications prennent effet.
- l) Répétez l'étape 3j pour les gestionnaires de files d'attente éloignées, ainsi que les paramètres équivalents pour les applications qui se connectent au gestionnaire de files d'attente migré, notamment:
- Noms de connexion de canal (attributCONNAME).
 - Règles d'authentification de canal qui restreignent les connexions entrantes du gestionnaire de files d'attente en fonction de son adresse IP ou de son nom d'hôte.

- Tables de définition de canal du client (CCDT), paramètres de nom de domaine (DNS), routage réseau ou informations de connexion équivalentes.
- m) Effectuez une reprise en ligne gérée du gestionnaire de files d'attente sur chaque noeud de gestionnaire de files d'attente de données répliquées pour vous assurer que la configuration requise a été correctement établie. Voir [«Définition de l'emplacement préféré d'un gestionnaire de files d'attente de données répliquées»](#), à la page 636.

Redimensionnement du système de fichiers pour un gestionnaire de files d'attente de données répliquées à haute disponibilité

Pour redimensionner le système de fichiers d'un gestionnaire de files d'attente de données répliquées (RDQM) à haute disponibilité existant, sauvegardez ses données persistantes, puis restaurez les données dans un gestionnaire de files d'attente de données répliquées nouvellement créé ayant le même nom mais un système de fichiers de taille différente.

Pourquoi et quand exécuter cette tâche

Les gestionnaires de files d'attente de données répliquées à haute disponibilité requièrent un volume logique dédié (système de fichiers) et la configuration de la réplication de disque et du contrôle à haute disponibilité. Ces composants sont configurés uniquement lorsqu'un nouveau gestionnaire de files d'attente est créé. Le système de fichiers ne peut pas être redimensionné une fois qu'il a été créé car il doit avoir la même taille sur chaque noeud. Pour redimensionner le système de fichiers d'un gestionnaire de files d'attente de données répliquées (RDQM) existant, vous pouvez sauvegarder ses données persistantes, puis restaurer les données dans un gestionnaire de files d'attente de données répliquées nouvellement créé qui a le même nom mais un système de fichiers d'une taille différente. Cette procédure permet de conserver la configuration, l'état et les messages persistants du gestionnaire de files d'attente au moment de la création de la sauvegarde.

Procédure

1. Sauvegardez le gestionnaire de files d'attente de données répliquées existant sur le noeud de gestionnaire de files d'attente de données répliquées principal:
 - a) Déterminez le noeud de gestionnaire de files d'attente de données répliquées principal pour le gestionnaire de files d'attente. Pour plus d'informations sur la détermination du noeud principal, voir [rdqmstatus \(display RDQM status\)](#).
 - b) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, si le gestionnaire de files d'attente de données répliquées est démarré, arrêtez-le à l'aide de la commande **endmqm -w** ou **endmqm -i**.
 - c) Déterminez l'emplacement du répertoire de données du gestionnaire de files d'attente en affichant le fichier de configuration IBM MQ, `mqs.ini`. Sous Linux, ce fichier se trouve dans le répertoire `/var/mqm`. Pour plus d'informations sur `mqs.ini`, voir [«Fichier de configuration IBM MQ, mqs.ini»](#), à la page 98.

Recherchez la section `QueueManager` du gestionnaire de files d'attente dans le fichier. Le répertoire de données du gestionnaire de files d'attente est la valeur de la clé nommée `DataPath`. Pour plus d'informations sur la section `QueueManager`, voir [«Section QueueManager du fichier mqs.ini»](#), à la page 109.

- d) Créez une sauvegarde du répertoire de données du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande **tar**. Par exemple, pour sauvegarder le répertoire de données d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (`.`) unique:

```
tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
```

- e) Déterminez l'emplacement du répertoire de journaux du gestionnaire de files d'attente en affichant le IBM MQ fichier de configuration du gestionnaire de files d'attente `qm.ini`. Ce fichier se trouve

dans le répertoire de données du gestionnaire de files d'attente. Pour plus d'informations sur le fichier, voir «Fichiers de configuration du gestionnaire de files d'attente, qm.ini», à la page 111.

Le répertoire de journaux du gestionnaire de files d'attente est défini en tant que valeur de la clé LogPath dans la section Journal. Pour plus d'informations sur la strophe, voir «Section de journal du fichier qm.ini», à la page 147.

- f) Créez une sauvegarde du répertoire de journaux du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande **tar**. Par exemple, pour sauvegarder le répertoire de journaux d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (.) unique:

```
tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
```

- g) Supprimez le gestionnaire de files d'attente de données répliquées existant.

2. Restaurez le gestionnaire de files d'attente avec un système de fichiers de la taille requise:

- a) Créez un gestionnaire de files d'attente de données répliquées (RDQM) portant le même nom que le gestionnaire de files d'attente que vous avez sauvegardé. Vérifiez que le système de fichiers alloué au gestionnaire de files d'attente de données répliquées par **crtmqm** correspond à la taille requise et qu'il est suffisamment grand pour contenir les données, les journaux principaux et les journaux secondaires du gestionnaire de files d'attente existant, plus un espace supplémentaire pour une extension ultérieure. Pour plus d'informations sur la création d'un gestionnaire de files d'attente de données répliquées, voir «Créer un gestionnaire de files d'attente de données répliquées», à la page 626.
- b) Déterminez le noeud de gestionnaire de files d'attente de données répliquées principal pour le gestionnaire de files d'attente. Pour plus d'informations sur la détermination du noeud principal, voir `rdqmstatus (display RDQM status)`.
- c) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, si le gestionnaire de files d'attente de données répliquées est démarré, arrêtez-le à l'aide de la commande **endmqm -w** ou **endmqm -i**.
- d) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, déterminez le nouvel emplacement des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées (utilisez les méthodes décrites dans les étapes 1c et 1e).
- e) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, supprimez le contenu des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées, mais pas les répertoires eux-mêmes.
- f) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, restaurez la sauvegarde du répertoire de données du gestionnaire de files d'attente dans le répertoire de données vide du nouveau gestionnaire de files d'attente de données répliquées, en vous assurant que la propriété des fichiers et les droits d'accès sont préservés. Si la sauvegarde a été créée à l'aide de l'exemple de commande **tar** à l'étape 1d, la commande suivante peut être utilisée par le superutilisateur pour la restaurer:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

- g) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, restaurez la sauvegarde du répertoire de journaux du gestionnaire de files d'attente dans le répertoire de journaux vide du nouveau gestionnaire de files d'attente de données répliquées, en veillant à préserver la propriété et les droits d'accès aux fichiers. Si la sauvegarde a été créée à l'aide de l'exemple de commande **tar** à l'étape 1f, la commande suivante peut être utilisée par l'utilisateur root pour la restaurer:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

- h) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, éditez le fichier de configuration de gestionnaire de files d'attente restauré, `qm.ini`, dans le répertoire de données du nouveau gestionnaire de files d'attente de données répliquées. Mettez à jour la valeur de la clé LogPath dans la section Log afin de spécifier le répertoire de journaux pour le nouveau

gestionnaire de files d'attente de données répliquées que vous avez déterminé à l'étape 2d. Passez en revue les autres chemins de fichier définis dans le fichier de configuration et mettez-les à jour si nécessaire. Par exemple, vous devrez peut-être mettre à jour les chemins suivants:

- Chemin d'accès aux fichiers journaux d'erreurs générés par les services de message de diagnostic.
 - Chemin d'accès aux exits requis par le gestionnaire de files d'attente.
 - Chemin d'accès aux fichiers de commutation de chargement si le gestionnaire de files d'attente est un coordinateur de transaction XA.
- i) Vérifiez que le gestionnaire de files d'attente est affiché par la commande **dspmq** et que son statut est signalé comme étant arrêté. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente à haute disponibilité de gestionnaire de files d'attente de données répliquées:

```
$ dspmq -o status -o ha
QMNAME(QM1) STATUS(Ended normally) HA(Replicated)
```

- j) Vérifiez que les données du gestionnaire de files d'attente restaurées ont été répliquées sur les noeuds de gestionnaire de files d'attente de données répliquées secondaires à l'aide de la commande **rdqmstatus** pour afficher le statut du gestionnaire de files d'attente. Le statut de la haute disponibilité doit être signalé comme Normal sur chaque noeud. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente à haute disponibilité de gestionnaire de files d'attente de données répliquées:

```
$ rdqmstatus -m QM1
Node: mqhavam10-adm
Queue manager status: Ended normally
Queue manager file system: 50MB used, 0.2GB
allocated [42%]
HA role: Primary
HA status: Normal
HA control: Disabled
HA current location: This node
HA preferred location: This node
HA floating IP interface: None
HA floating IP address: None
Node: mqhavam11-adm
HA status: Normal
Node: mqhavam12-adm
HA status: Normal
```

- k) Démarrez le gestionnaire de files d'attente sur le noeud RDQM principal.
- l) Effectuez une reprise en ligne gérée du gestionnaire de files d'attente sur chaque noeud de gestionnaire de files d'attente de données répliquées pour vous assurer que la configuration requise a été correctement établie. Voir [«Définition de l'emplacement préféré d'un gestionnaire de files d'attente de données répliquées»](#), à la page 636.

Stockage du statut de l'application persistante

Vous pouvez stocker des informations de statut persistantes relatives aux applications ainsi que d'autres données de gestionnaire de files d'attente.

Chaque gestionnaire de files d'attente IBM MQ dispose d'un système de fichiers dédié pour son état persistant, qui inclut à la fois ses données de file d'attente et son journal de reprise. Dans une configuration de gestionnaire de files d'attente de données répliquées, le système de fichiers est sauvegardé par un volume logique qui est répliqué entre les systèmes Linux (noeuds). Le système de fichiers inclut un répertoire `userdata` que vous pouvez utiliser pour stocker les informations de statut persistantes de vos applications. Par conséquent, lorsqu'un gestionnaire de files d'attente de données répliquées se déplace pour s'exécuter sur un autre noeud de votre configuration de gestionnaire de files d'attente de données répliquées, vous disposez d'un contexte d'application et d'un contexte de gestionnaire de files d'attente. Voir [Contenu d'annuaire sur les systèmes Unix et Linux](#).

Si vous choisissez de stocker l'état de l'application dans le répertoire `userdata`, vous devez savoir que les données écrites à cet emplacement peuvent consommer l'espace disque disponible alloué au gestionnaire de files d'attente. Vous devez vous assurer que l'espace disque disponible reste suffisant

pour que le gestionnaire de files d'attente puisse écrire des données de file d'attente, des journaux et d'autres informations d'état persistantes.

Le répertoire `userdata` est propriétaire de l'utilisateur et du groupe `mqm` et est accessible en lecture à tous, de sorte que les utilisateurs puissent y accéder sans avoir à faire partie du groupe d'administrateurs IBM MQ (c'est-à-dire `mqm`). Vous ne pouvez pas modifier les droits du répertoire `userdata`, mais vous pouvez y créer du contenu avec la propriété et les droits dont vous avez besoin.

Lors d'une reprise en ligne du gestionnaire de files d'attente de données répliquées, le gestionnaire de files d'attente est arrêté et son système de fichiers est démonté sur son noeud de gestionnaire de files d'attente de données répliquées en cours. Le système de fichiers est ensuite monté et le gestionnaire de files d'attente est redémarré sur un autre noeud dans la configuration du gestionnaire de files d'attente de données répliquées. Un système de fichiers ne peut pas être démonté si un processus a un descripteur ouvert pour l'un de ses fichiers. Pour s'assurer que la reprise en ligne d'un gestionnaire de files d'attente peut être terminée, si le système de fichiers du gestionnaire de files d'attente ne peut pas être démonté, les processus ayant un descripteur de fichier ouvert reçoivent un signal `SIGTERM`, suivi d'un signal `SIGKILL` si les descripteurs ouverts ne sont pas libérés. Vos applications doivent être conçues pour répondre correctement à `SIGTERM`. Si des applications ou des processus sont configurés en tant que service de gestionnaire de files d'attente, lors d'une reprise en ligne gérée, ils peuvent être arrêtés lors de l'arrêt du gestionnaire de files d'attente avant que le système de fichiers ne soit démonté. Si une application ou un processus n'est pas configuré en tant que service de gestionnaire de files d'attente ou qu'une reprise en ligne non gérée se produit, telle qu'une perte de quorum, il est probable que des signaux seront envoyés pour libérer le système de fichiers.

Linux Définition de l'emplacement préféré d'un gestionnaire de files d'attente de données répliquées

L'emplacement préféré d'un gestionnaire de files d'attente de données répliquées (RDQM) identifie le noeud sur lequel le gestionnaire de files d'attente de données répliquées doit s'exécuter si ce noeud est disponible.

Pourquoi et quand exécuter cette tâche

L'emplacement préféré est le nom du noeud sur lequel Pacemaker doit exécuter le gestionnaire de files d'attente lorsque le groupe HA est dans un état normal (tous les noeuds et les connexions disponibles). L'emplacement préféré est initialisé avec le nom du noeud principal lors de la création du gestionnaire de files d'attente. Vous pouvez exécuter les commandes pour définir l'emplacement préféré sur l'un des trois noeuds. Vous devez être un utilisateur appartenant aux groupes `mqm` et `haclient`.

Procédure

- Pour affecter le noeud local ou spécifié comme emplacement préféré du gestionnaire de files d'attente nommé, entrez la commande suivante:

```
rdqmadm -p -m qmname [ -n nodename [,nodename ]
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées pour lequel vous spécifiez l'emplacement préféré et *nodename* est éventuellement le nom du noeud préféré.

Si le groupe HA est dans un état normal et que l'emplacement préféré n'est pas le noeud principal en cours, le gestionnaire de files d'attente est arrêté et redémarré sur le nouvel emplacement préféré. Vous pouvez spécifier une liste de deux noms de noeud séparés par des virgules pour affecter une deuxième préférence d'emplacement préféré.

- Pour effacer l'emplacement préféré afin que le gestionnaire de files d'attente ne revienne pas automatiquement à un noeud lorsqu'il est restauré, entrez la commande suivante:

```
rdqmadm -p -m qmname -d
```

Référence associée

[rdqmadm \(administrer un cluster de gestionnaires de files d'attente de données répliquées\)](#)

Création et suppression d'une adresse IP flottante

Une adresse IP flottante permet à un client d'utiliser la même adresse IP pour un gestionnaire de files d'attente de données répliquées (RDQM), quel que soit le noeud du groupe à haute disponibilité sur lequel il s'exécute. (L'utilisation d'une adresse IP flottante est facultative.)

Pourquoi et quand exécuter cette tâche

Vous pouvez créer ou supprimer une adresse IP flottante à l'aide de la commande **rdqmint**. L'adresse flottante est liée à une interface physique nommée sur le noeud principal du gestionnaire de files d'attente de données répliquées. Si le gestionnaire de files d'attente de données répliquées bascule et qu'un noeud différent devient le noeud principal, l'adresse IP flottante est liée à une interface du même nom sur le nouveau noeud principal. Les interfaces physiques sur les trois noeuds doivent appartenir au même sous-réseau que l'adresse IP flottante. Le diagramme suivant illustre l'utilisation d'une adresse IP flottante.

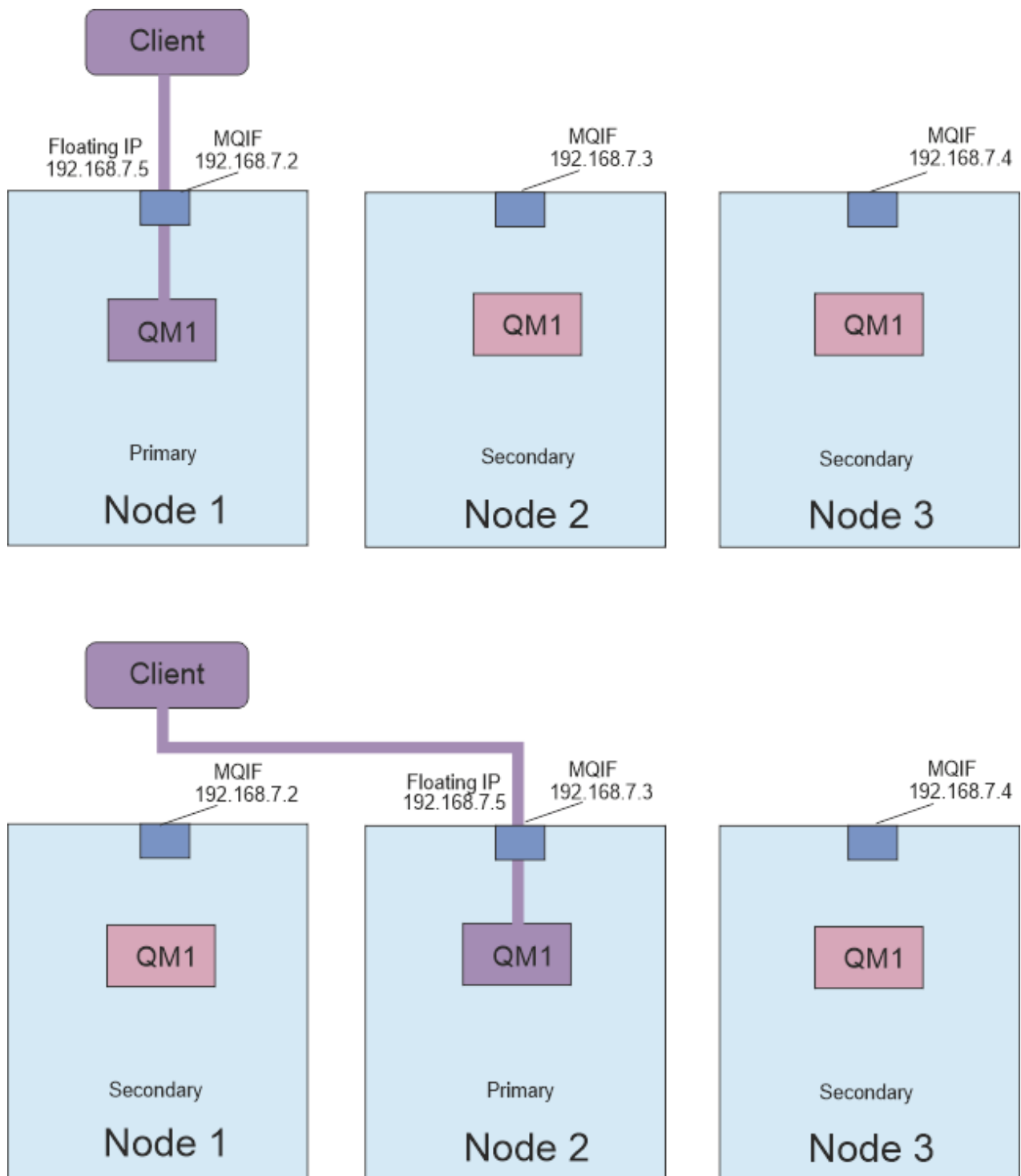


Figure 80. Adresse IP flottante

Vous devez être un utilisateur des groupes `mqm` et `haclient` pour exécuter la commande `rdqmint`. Vous pouvez créer ou supprimer l'adresse IP flottante sur le noeud principal du gestionnaire de files d'attente de données répliquées (RDQM) ou sur l'un des noeuds secondaires.

Remarque : Vous ne pouvez pas utiliser la même adresse IP flottante pour plusieurs gestionnaires de files d'attente de données répliquées, l'adresse IP flottante de chaque gestionnaire de files d'attente de données répliquées doit être unique.

Procédure

- Pour créer une adresse IP flottante pour un gestionnaire de files d'attente de données répliquées, entrez la commande suivante:

```
rdqmint -m qmname -a -f ipv4address -l interfacename
```

où :

NomGestFile

Nom du gestionnaire de files d'attente de données répliquées pour lequel vous créez l'adresse IP flottante.

ipv4address

Adresse IP flottante au format ipv4 .

L'adresse IP flottante doit être une adresse IPv4 valide qui n'est pas déjà définie sur un noeud à haute disponibilité et elle doit appartenir au même sous-réseau que les adresses IP statiques définies pour l'interface locale.

interfaceName

Nom de l'interface physique sur le noeud principal auquel établir la liaison.

Exemple :

```
rdqmint -m QM1 -a -f 192.168.7.5 -l MQIF
```

- Pour supprimer une adresse IP flottante existante, entrez la commande suivante:

```
rdqmint -m qmname -d
```

Référence associée

[rdqmint \(ajout ou suppression d'une adresse IP flottante pour un gestionnaire de files d'attente de données répliquées\)](#)

Linux

Démarrage, arrêt et affichage de l'état d'un gestionnaire de files d'attente de données répliquées à haute disponibilité

Vous utilisez des variantes des commandes de contrôle IBM MQ standard pour démarrer, arrêter et afficher l'état en cours d'un gestionnaire de files d'attente de données répliquées (RDQM).

Pourquoi et quand exécuter cette tâche

Vous devez exécuter les commandes qui démarrent, arrêtent et affichent l'état en cours d'un gestionnaire de files d'attente de données répliquées (RDQM) en tant qu'utilisateur appartenant aux groupes mqm et haclient .

Vous devez exécuter les commandes pour démarrer et arrêter un gestionnaire de files d'attente sur le noeud principal de ce gestionnaire de files d'attente.

Procédure

- Pour démarrer un gestionnaire de files d'attente de données répliquées, entrez la commande suivante sur le noeud principal du gestionnaire de files d'attente de données répliquées:

```
strmqm qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées que vous souhaitez démarrer.

Le gestionnaire de files d'attente de données répliquées est démarré et Pacemaker commence à gérer le gestionnaire de files d'attente de données répliquées. Vous devez spécifier l'option -ns avec `strmqm` si vous souhaitez spécifier d'autres options `strmqm` .

- Pour arrêter un gestionnaire de files d'attente de données répliquées, entrez la commande suivante sur le noeud principal du gestionnaire de files d'attente de données répliquées:

```
endmqm qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées que vous souhaitez arrêter.

Pacemaker cesse de gérer le gestionnaire de files d'attente de données répliquées, puis le gestionnaire de files d'attente de données répliquées est arrêté. Tous les autres paramètres **endmqm** peuvent être utilisés lors de l'arrêt d'un gestionnaire de files d'attente de données répliquées.

- Pour afficher l'état d'un gestionnaire de files d'attente de données répliquées, entrez la commande suivante:

```
dspmq
```

Les informations d'état qui sont générées varient selon que vous exécutez la commande sur le noeud principal ou secondaire du gestionnaire de files d'attente de données répliquées. S'il est exécuté sur le noeud principal, l'un des messages de statut normal renvoyés par **dspmq** s'affiche. Si vous exécutez la commande sur un noeud secondaire, l'état `running elsewhere` s'affiche. Par exemple, si **dspmq** est exécuté sur le noeud RDQM7, les informations suivantes peuvent être renvoyées:

```
QMNAME (RDQM8)          STATUS(Running elsewhere)
QMNAME (RDQM9)          STATUS(Running elsewhere)
QMNAME (RDQM7)          STATUS(Running)
```

Si le noeud principal n'est pas disponible ou si **dspmq** est exécuté par un utilisateur qui n'est pas `root` ou membre du groupe `haclient`, l'état `Unavailable` est signalé. Exemple :

```
QMNAME (RDQM8)          STATUS(Unavailable)
QMNAME (RDQM9)          STATUS(Unavailable)
QMNAME (RDQM7)          STATUS(Unavailable)
```

Vous pouvez entrer la commande **dspmq -o ha** (ou **dspmq -o HA**) pour afficher la liste des gestionnaires de files d'attente connus d'un noeud et indiquer s'il s'agit ou non de gestionnaires de files d'attente de données répliquées, par exemple:

```
dspmq -o ha
QMNAME (RDQM8)          HA(Replicated)
QMNAME (RDQM9)          HA(Replicated)
QMNAME (RDQM7)          HA(Replicated)
QMNAME (QM7)            HA()
```

Référence associée

[dspmq \(affichage des gestionnaires de files d'attente\)](#)

[endmqm \(arrêt d'un gestionnaire de files d'attente\)](#)

[strmqm \(démarrage d'un gestionnaire de files d'attente\)](#)

Actions de ressource ayant échoué

Les actions de ressource ayant échoué se produisent lorsque le composant Pacemaker d'une configuration à haute disponibilité de gestionnaire de files d'attente de données répliquées rencontre un problème avec une ressource sur l'un des noeuds d'un groupe à haute disponibilité.

La solution à haute disponibilité des gestionnaires de files d'attente de données répliquées utilise Pacemaker pour la surveillance et la gestion des ressources (voir «Haute disponibilité de RDQM», à la page 616). Si Pacemaker rencontre une erreur lors de l'exécution d'une opération sur une ressource d'un noeud, il enregistre ces informations à l'aide d'une action de ressource ayant échoué. Certaines actions de ressource ayant échoué empêchent l'exécution de la ressource et doivent être effacées pour que Pacemaker puisse redémarrer la ressource.

Vous pouvez utiliser la commande **rdqmstatus -m** pour déterminer si des actions de ressource ayant échoué empêchent un gestionnaire de files d'attente de démarrer sur un ou plusieurs noeuds.

Vous pouvez ensuite utiliser la commande **rdqmstatus -m qmname -a** pour afficher les détails des actions de ressource ayant échoué qui sont associées à un gestionnaire de files d'attente. Suivez cette action à l'aide de la commande **rdqmclean** pour effacer ces actions de ressource ayant échoué et ainsi libérer les ressources restreintes. (Vous devez également prendre des mesures pour résoudre les problèmes qui ont provoqué l'échec de l'action de ressource en premier lieu.)

Les ressources suivantes sont contrôlées par Pacemaker dans une configuration à haute disponibilité de gestionnaire de files d'attente de données répliquées (RDQM HA) et peuvent faire l'objet d'actions de ressource ayant échoué:

- Gestionnaire de files d'attente
- IP flottante
- Contrôle RDQM
- Système de fichiers
- Réplication de reprise après incident (DRBD)
- Réplication à haute disponibilité (DRBD)

Chaque type de ressource peut être sujet aux types d'échec suivants:

Lâche

Les échecs logiciels sont transitoires et Pacemaker continue de tenter de récupérer la ressource jusqu'à ce qu'elle arrive à expiration ou qu'elle soit arrêtée.

Strict

Une erreur matérielle nécessite une intervention administrative. Les erreurs matérielles empêchent la ressource de s'exécuter sur un noeud particulier.

Erreur bloquante

Une erreur fatale nécessite une intervention administrative. Des erreurs fatales empêchent la ressource de s'exécuter sur n'importe quel noeud.

Voir [«Affichage du statut des groupes RDQM et HA»](#), à la page 641 pour des exemples de statut incluant les actions de file d'attente de ressources ayant échoué.

Vous pouvez utiliser la commande **rdqmclean** pour effacer toutes les actions de ressource ayant échoué associées à un gestionnaire de files d'attente spécifié ou toutes les actions de ressource ayant échoué dans la configuration à haute disponibilité des gestionnaires de files d'attente de données répliquées.

Remarque : Certaines actions de ressource ayant échoué n'entraînent pas le blocage du gestionnaire de files d'attente sur un noeud. Par exemple, après un arrêt inattendu du gestionnaire de files d'attente, Pacemaker tente de redémarrer le gestionnaire de files d'attente sur le noeud sur lequel il a été détecté qu'il n'était pas en cours d'exécution. Si le démarrage aboutit, le gestionnaire de files d'attente n'est pas bloqué sur le noeud. La seule façon de prendre connaissance de l'action de ressource ayant échoué dans ce cas consiste à exécuter **rdqmstatus -m qmname -a**.

Tâches associées

[«Affichage du statut des groupes RDQM et HA»](#), à la page 641

Vous pouvez afficher le statut du groupe à haute disponibilité et des gestionnaires de files d'attente de données répliquées (RDQM) individuels.

Référence associée

[rdqmclean](#)

[rdqmstatus](#)

Affichage du statut des groupes RDQM et HA

Vous pouvez afficher le statut du groupe à haute disponibilité et des gestionnaires de files d'attente de données répliquées (RDQM) individuels.

Pourquoi et quand exécuter cette tâche

Vous utilisez la commande **rdqmstatus** pour afficher le statut des gestionnaires de files d'attente de données répliquées (RDQM) individuels et du groupe à haute disponibilité dans son ensemble.

Le statut récapitulatif d'un noeud affiche également des informations sur le module de noyau DRBD sur lequel repose le gestionnaire de files d'attente de données répliquées. Lorsque vous mettez à niveau un gestionnaire de files d'attente de données répliquées (RDQM), il est important de s'assurer que la version correcte du module de noyau DRBD est installée pour la version du noyau RHEL qui s'exécute sur le système. Le statut affiche la version du noyau du système d'exploitation, la version du noyau pour laquelle le module DRBD a été généré, la version DRBD et le statut de chargement du module de noyau DRBD.

Vous devez être un utilisateur des groupes `mqm` et `haclient` pour exécuter la commande **rdqmstatus**. Vous pouvez exécuter la commande sur l'un des trois noeuds.

Procédure

- Pour afficher le statut récapitulatif d'un noeud et des gestionnaires de files d'attente de données répliquées qui font partie de la configuration à haute disponibilité:

```
rdqmstatus
```

L'identité du noeud sur lequel vous avez exécuté la commande sur, le noyau et les détails DRBD de ce noeud, et le statut des gestionnaires de files d'attente de données répliquées dans la configuration à haute disponibilité sont affichés, par exemple:

```
Node:                               mqhavam07.exampleco.com
OS kernel version:                  5.14.0-362.18.1
DRBD OS kernel version:             5.14.0-362.18.1
DRBD version:                       9.2.7
DRBD kernel module status:          Loaded

Queue manager name:                 RDQM8
Queue manager status:               Running elsewhere
HA current location:                mqhavam08.exampleco.com
HA preferred location:              mqhavam08.exampleco.com
HA blocked location:                None

Queue manager name:                 RDQM9
Queue manager status:               Running elsewhere
HA current location:                mqhavam09.exampleco.com
HA preferred location:              mqhavam09.exampleco.com
HA blocked location:                None

Queue manager name:                 RDQM7
Queue manager status:               Running
HA current location:                This node
HA preferred location:              This node
HA blocked location:                None
```

Le statut du module de noyau DRBD est l'une des valeurs suivantes:

Chargé

Indique que le module DRBD a été chargé.

Partiellement chargé

Peut se produire lorsque le module DRBD a été chargé, mais ne fonctionne pas correctement en raison d'une non-concordance.

Non chargé

Le module DRBD n'est pas chargé. Cela peut être affiché dans une configuration nouvellement installée, lorsqu'aucun gestionnaire de files d'attente de données répliquées n'a encore été créé.

Non installé

Indique que le module DRBD n'est pas installé, ou que IBM MQ n'a pas pu déterminer la version du noyau du système d'exploitation du module DRBD.

Version déjà installée encore chargée

Ce statut peut se produire si un nouveau module DRBD est installé alors que le module DRBD existant est en cours d'exécution (c'est-à-dire qu'un gestionnaire de files d'attente RDQM est en cours d'exécution). Le module nouvellement installé est signalé dans le statut, mais il ne s'agit pas du module en cours d'exécution.

- Pour afficher le statut des trois noeuds du groupe HA, entrez la commande suivante:

```
rdqmstatus -n
```

Le statut en ligne ou hors ligne de chaque noeud est signalé. Exemple :

```
Node mqha04(mqhavm04.example.com) is online
Node mqha05(mqhavm05.example.com) is offline
Node mqha06(mqhavm06.example.com) is online
```

- Pour afficher le statut d'un gestionnaire de files d'attente particulier sur tous les noeuds du groupe à haute disponibilité, entrez la commande suivante:

```
rdqmstatus -m qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées dont vous souhaitez afficher le statut. Le statut du gestionnaire de files d'attente de données répliquées sur le noeud en cours s'affiche, suivi d'un récapitulatif du statut des deux autres noeuds du point de vue du noeud en cours.

- Pour afficher le statut d'un gestionnaire de files d'attente particulier sur tous les noeuds du groupe à haute disponibilité, y compris les détails des actions sur les ressources ayant échoué, entrez la commande suivante:

```
rdqmstatus -m qmname -a
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées dont vous souhaitez afficher le statut. Le statut du gestionnaire de files d'attente de données répliquées sur le noeud en cours s'affiche, suivi d'un récapitulatif du statut des deux autres noeuds du point de vue du noeud en cours. Elle est suivie de détails sur les actions de ressource ayant échoué associées au gestionnaire de files d'attente de données répliquées (RDQM).

- Le tableau suivant récapitule les informations sur le noeud en cours qui peuvent être renvoyées par la commande `rdqmstatus -m qmname` pour un gestionnaire de files d'attente de données répliquées.

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Nom du noeud	<i>nodeName</i>	Toujours affiché
Etat de gestionnaire de files d'attente	En cours de fonctionnement Exécuté ailleurs Terminé Non disponible	Toujours affiché
UC	<i>n.nn%</i>	Affiché uniquement lorsque le noeud en cours a un rôle principal (c'est-à-dire que le gestionnaire de files d'attente de données répliquées est en cours d'exécution sur ce noeud)

Tableau 33. Statut en cours du noeud (suite)

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Mémoire	<i>nnn</i> Mo utilisés, <i>y.y</i> Go alloués	Affiché uniquement lorsque le noeud en cours a un rôle principal (c'est-à-dire que le gestionnaire de files d'attente de données répliquées est en cours d'exécution sur ce noeud)
Système de fichiers du gestionnaire de files d'attente	<i>nnn</i> Mo utilisés, <i>y.y</i> Go alloués [z%]	Affiché uniquement lorsque le noeud en cours a un rôle principal (c'est-à-dire que le gestionnaire de files d'attente de données répliquées est en cours d'exécution sur ce noeud)
Rôle HA	Secondaire principal inconnu	Toujours affiché
Statut de la haute disponibilité	Tous les noeuds en veille Ce noeud en veille Noeuds distants en veille Mixte <i>état des noeuds distants</i>	Tous les noeuds en veille Noeud en cours en veille Les deux noeuds distants en veille Statut différent pour chaque noeud distant (voir le tableau suivant pour le statut individuel) Même statut pour les deux noeuds distants (voir le tableau suivant pour toutes les valeurs)
Contrôle de haute disponibilité	Activé Désactivé Inconnu	Toujours affiché. Indique si le gestionnaire de files d'attente de données répliquées est sous le contrôle Pacemaker
Emplacement préféré de la haute disponibilité	Aucun Ce noeud Inconnu <i>nodeName</i>	Toujours affiché

Tableau 33. Statut en cours du noeud (suite)

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Emplacement bloqué de la haute disponibilité	Aucun-Le gestionnaire de files d'attente n'est pas bloqué sur les noeuds Ce noeud-Le gestionnaire de files d'attente ne peut pas s'exécuter sur le noeud en cours en raison d'une ou de plusieurs actions de ressource ayant échoué <i>nodename</i> -Le gestionnaire de files d'attente ne peut pas s'exécuter sur <i>nodename</i> en raison d'une ou de plusieurs actions de ressource ayant échoué <i>nodename1, nodename2</i> -Le gestionnaire de files d'attente ne peut pas s'exécuter sur <i>nodename1</i> et <i>nodename2</i> en raison d'une ou de plusieurs actions de ressource ayant échoué Tous les noeuds-Le gestionnaire de files d'attente ne peut pas s'exécuter sur tous les noeuds en raison d'une ou de plusieurs actions de ressource ayant échoué	Toujours affiché
Interface IP flottante à haute disponibilité	<i>interface_name</i>	Toujours affiché
Adresse IP flottante haute disponibilité	<i>IPV4_address</i>	Toujours affiché

Le tableau suivant récapitule les informations renvoyées par la commande `rdqmstatus -m qmname` pour les autres noeuds du groupe HA.

Tableau 34. Autre statut de noeud

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Nom du noeud	<i>nodename</i>	Toujours affiché
Statut de la haute disponibilité	Normal Synchronisation en cours Eloigné non disponible Incohérent Suspendu Noeud éloigné en veille Inconnu	Les noeuds sont synchronisés les uns avec les autres Synchronisation avec le noeud distant Impossible de communiquer avec le noeud distant Non synchronisé avec le noeud distant et non synchronisé Réplication en pause Noeud éloigné en veille
Synchronisation de la haute disponibilité en cours	<i>n.n%</i>	S'affiche lorsque la synchronisation est en cours et que la commande est exécutée en tant que <code>root</code>

Tableau 34. Autre statut de noeud (suite)

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Temps de synchronisation estimé pour la haute disponibilité	aaaa-mm-jj hh:mm:ss.nnn	S'affiche lorsque la synchronisation est en cours
Données de haute disponibilité non synchronisées	nKo	S'affiche lorsque le noeud distant est indisponible ou incohérent
Dernière synchronisation de la haute disponibilité	aaaa-mm-jj hh:mm:ss.nnn	S'affiche lorsque les données de haute disponibilité ne sont pas synchronisées (après la synchronisation initiale). Indique l'heure et la date de la dernière synchronisation des données.

Exemple

Exemple de statut normal sur le noeud principal:

```

Node:                               mqhavam07.exampleco.com
Queue manager status:              Running
CPU:                               0.00
Memory:                            123MB
Queue manager file system:         606MB used, 1.0GB allocated [60%]
HA role:                           Primary
HA status:                          Normal
HA control:                         Enabled
HA current location:                This node
HA preferred location:              This node
HA preferred location:              This node
HA blocked location:                None
HA floating IP interface:           eth4
HA floating IP address:             192.0.2.4

Node:                               mqhavam08.exampleco.com
HA status:                          Normal

Node:                               mqhavam09.exampleco.com
HA status:                          Normal

```

Exemple de statut normal sur un noeud secondaire:

```

Node:                               mqhavam08.exampleco.com
Queue manager status:              Running elsewhere
HA role:                           Secondary
HA status:                          Normal
HA control:                         Enabled
HA current location:                mqhavam07.exampleco.com
HA preferred location:              mqhavam07.exampleco.com
HA blocked location:                None
HA floating IP interface:           eth4
HA floating IP address:             192.0.2.4

Node:                               mqhavam07.exampleco.com
HA status:                          Normal

Node:                               mqhavam09.exampleco.com
HA status:                          Normal

```

Exemple de statut sur le noeud principal lorsque la synchronisation est en cours:

```

Node:                               mqhavam07.exampleco.com
Queue manager status:              Running
CPU:                               0.53
Memory:                            124MB
Queue manager file system:         51MB used, 1.0GB allocated [5%]
HA role:                           Primary
HA status:                          Synchronization in progress
HA control:                         Enabled

```

```

HA current location:      This node
HA preferred location:   This node
HA blocked location:     None
HA floating IP interface: eth4
HA floating IP address:  192.0.2.4

Node:                    mqhavam08.exampleco.com
HA status:               Synchronization in progress
HA synchronization progress: 11.0%
HA estimated time to completion: 2017-09-06 14:55:05

Node:                    mqhavam09.exampleco.com
HA status:               Synchronization in progress
HA synchronization progress: 11.0%
HA estimated time to completion: 2017-09-06 14:55:06

```

Exemple de statut sur le noeud principal lorsque la synchronisation a été perdue:

```

Node:                    mqhavam07.exampleco.com
Queue manager status:   Running
CPU:                    0.53
Memory:                 124MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role:                Primary
HA status:              Mixed
HA control:             Enabled
HA current location:    This node
HA preferred location:  This node
HA blocked location:    None
HA floating IP interface: eth4
HA floating IP address: 192.0.2.4

Node:                    mqhavam08.exampleco.com
HA status:              Normal

Node:                    mqhavam09.exampleco.com
HA status:              Inconsistent
HA out of sync data:    15932KB
HA last in sync:       2017-09-06 14:55:06

```

Exemple de noeud principal présentant plusieurs états:

```

Node:                    mqhavam07.exampleco.com
Queue manager status:   Running
CPU:                    0.02
Memory:                 124MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role:                Primary
HA status:              Mixed
HA control:             Enabled
HA current location:    This node
HA preferred location:  This node
HA blocked location:    None
HA floating IP interface: eth4
HA floating IP address: 192.0.2.4

Node:                    mqhavam08.exampleco.com
HA status:              Normal

Node:                    mqhavam09.exampleco.com
HA status:              Inconsistent

```

Exemple de noeud principal montrant les actions de ressource ayant échoué:

```

Node:                    mqhavam07.exampleco.com
Queue manager status:   Running
CPU:                    0.00%
Memory:                 123MB
Queue manager file system: 606MB used, 1.0GB allocated [60%]
HA role:                Primary
HA status:              Normal
HA control:             Enabled
HA current location:    This node
HA preferred location:  mqhavam08.exampleco.com
HA blocked location:    mqhavam08.exampleco.com
HA floating IP interface: eth4
HA floating IP address: 192.0.2.4

```

```

Node:                               mqhavam08.exampleco.com
HA status:                           Normal

Node:                               mqhavam09.exampleco.com
HA status:                           Normal

Failed resource action:              Start
Resource type:                       Filesystem
Failure node:                         mqhavam08.exampleco.com
Failure time:                         2017-09-06 12:00:00
Failure reason:                       Couldn't find directory [/var/mqm/vols/qmname] to use
as a mount point
Blocked location:                     mqhavam08.exampleco.com

```

Ce statut indique que Pacemaker n'a pas pu démarrer le système de fichiers sur le noeud mqhavam08.exampleco.com à 12:00:00. Cette action sur les ressources ayant échoué signifie que le gestionnaire de files d'attente ne peut pas s'exécuter sur mqhavam08.exampleco.com. Une fois que le problème sous-jacent à l'origine de l'échec de l'action de ressource a été résolu, exécutez la commande **rdqmclean** pour effacer l'action ayant échoué afin que Pacemaker puisse relancer l'action (si nécessaire).

Exemple de statut récapitulatif montrant une non-concordance entre la version du noyau du système d'exploitation (RHEL 9.3) et le module de noyau DRBD (ciblé pour RHEL 9.2). Même si les rapports de statut indiquent que le module de noyau DRBD est chargé et que le gestionnaire de files d'attente est en cours d'exécution, vous devez mettre à jour le module de noyau DRBD avec la version ciblée pour le noyau de système d'exploitation en cours d'exécution dans cette situation.

```

Node:                               mqhavam07.exampleco.com
OS kernel version:                   5.14.0-362.18.1
DRBD OS kernel version:              5.14.0-284.11.1
DRBD version:                        9.2.7+ptf.14
DRBD kernel module status:           Loaded

Queue manager name:                  RDQM7
Queue manager status:                Running
HA current location:                 This node
HA preferred location:               This node
HA blocked location:                 None

```

Exemple de statut récapitulatif montrant une non-concordance entre la version de noyau du système d'exploitation (RHEL 8.10) et le module de noyau DRBD (destiné à RHEL 8.8). Dans cet exemple, la non-concordance de version est plus grave et le chargement du module de noyau DRBD échoue. En conséquence, le gestionnaire de files d'attente ne parvient pas à démarrer sur son noeud préférentiel et son statut de haute disponibilité dans Unknown. Pour résoudre cet incident, le module de noyau DRBD doit être mis à jour avec la cible de version du noyau du système d'exploitation en cours d'exécution.

```

Node:                               mqhavam57.exampleco.com
OS kernel version:                   4.18.0-553
DRBD OS kernel version:              4.18.0-477
DRBD version:                        9.2.7+ptf.14
DRBD kernel module status:           Partially loaded

Queue manager name:                  QM2
Queue manager status:                Running elsewhere
HA status:                           Unknown
HA current location:                 mqhavam58.exampleco.com
HA preferred location:               This node
HA blocked location:                 All nodes

```

Référence associée

 [rdqmstatus](#)

Changement des adresses IP dans les configurations à haute disponibilité

Si vous modifiez les adresses IP de l'une des interfaces dans une configuration à haute disponibilité, l'opération de haute disponibilité n'est plus disponible et le gestionnaire de files d'attente ne s'exécute pas sur le noeud sur lequel les adresses ont été modifiées.

Vous pouvez spécifier jusqu'à trois adresses IP pour l'opération de haute disponibilité dans le fichier `rdqm.ini`. Si vous avez déjà modifié les adresses du moniteur Pacemaker, vous devez les restaurer temporairement à leurs valeurs d'origine avant de suivre la procédure. Sinon, il n'est pas possible de supprimer le gestionnaire de files d'attente de données répliquées à haute disponibilité.

1. Supprimez la configuration à haute disponibilité sur chaque noeud. Vous retirez la haute disponibilité en sauvegardant les gestionnaires de files d'attente, puis en les supprimant (voir «[Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ](#)», à la page 737 et «[Suppression d'un RDQM à haute disponibilité](#)», à la page 628), puis en retirant le groupe à haute disponibilité lui-même (voir «[Suppression du cluster Pacemaker \(groupe HA\)](#)», à la page 626).
2. Recréez la configuration à haute disponibilité avec les nouvelles adresses IP. Voir «[Définition du cluster Pacemaker \(groupe HA\)](#)», à la page 623.
3. Recréez les gestionnaires de files d'attente à haute disponibilité et restaurez la sauvegarde (voir «[Créer un gestionnaire de files d'attente de données répliquées](#)», à la page 626 et «[Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ](#)», à la page 737).

Linux Remplacement d'un noeud défaillant dans une configuration à haute disponibilité

Si l'un des noeuds de votre groupe à haute disponibilité échoue, vous pouvez le remplacer.

Pourquoi et quand exécuter cette tâche

Les étapes à suivre pour remplacer un noeud dépendent du scénario:

- Si vous remplacez le noeud défaillant par un noeud de configuration identique, vous pouvez remplacer le noeud sans interrompre le groupe HA.
- Si le nouveau noeud a une configuration différente, vous devez supprimer, puis régénérer le groupe HA. Vous pouvez d'abord sauvegarder les gestionnaires de files d'attente à partir du noeud sur lequel ils s'exécutent, puis les restaurer après avoir régénéré le groupe à haute disponibilité.

Procédure

- Si le noeud de remplacement est configuré pour ressembler au noeud défaillant (même nom d'hôte, mêmes adresses IP, etc.), procédez comme suit sur le nouveau noeud:
 - a) Créez un fichier `rdqm.ini` qui correspond aux fichiers des autres noeuds, puis exécutez la commande `rdqmadm -c` (voir «[Définition du cluster Pacemaker \(groupe HA\)](#)», à la page 623).
 - b) Exécutez la commande `crtmqm -sxs qmanager` pour recréer chaque gestionnaire de files d'attente de données répliquées (voir «[Créer un gestionnaire de files d'attente de données répliquées](#)», à la page 626).
- Si le noeud de remplacement a une configuration différente de celle du noeud défaillant:
 - a) Si nécessaire, sauvegardez vos gestionnaires de files d'attente (voir «[Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ](#)», à la page 737).
 - b) Supprimez les gestionnaires de files d'attente de données répliquées des autres noeuds du groupe HA à l'aide de la commande `dlrmqm` (voir «[Suppression d'un RDQM à haute disponibilité](#)», à la page 628).
 - c) Annulez la configuration du cluster Pacemaker à l'aide de la commande `rdqmadm -u` (voir «[Suppression du cluster Pacemaker \(groupe HA\)](#)», à la page 626).
 - d) Reconfigurez le cluster Pacemaker, y compris les informations relatives au nouveau noeud, à l'aide de la commande `rdqmadm -c` (voir «[Définition du cluster Pacemaker \(groupe HA\)](#)», à la page 623).
 - e) Si nécessaire (c'est-à-dire si vous ne disposez pas d'un accès SSH aux autres noeuds), exécutez la commande `crtmqm -sxs qmanager` pour recréer chaque gestionnaire de files d'attente de données répliquées sur les autres noeuds (voir «[Créer un gestionnaire de files d'attente de données répliquées](#)», à la page 626).

- f) Exécutez la commande `crtmqm -sx qmanager` pour créer les gestionnaires de files d'attente sur le noeud de remplacement.
- g) Si nécessaire, restaurez les données et la configuration dans vos gestionnaires de files d'attente (voir «[Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ](#)», à la page 737).

MQ Adv.

Linux

Reprise après incident RDQM

Le gestionnaire de files d'attente de données répliquées (RDQM) est disponible sur un sous-ensemble de plateformes Linux et peut fournir une solution de reprise après incident.

Pour plus d'informations, voir [Software Product Compatibility Reports](#) .

Vous pouvez créer une instance principale d'un gestionnaire de files d'attente de reprise après incident s'exécutant sur un serveur et une instance secondaire du gestionnaire de files d'attente sur un autre serveur qui fait office de noeud de reprise. Les données sont dupliquées entre les instances du gestionnaire de files d'attente. Si vous perdez votre gestionnaire de files d'attente principal, vous pouvez définir manuellement votre instance secondaire comme instance principale et démarrer le gestionnaire de files d'attente, puis reprendre votre travail à l'endroit où vous vous étiez arrêté. Vous ne pouvez pas démarrer un gestionnaire de files d'attente lorsqu'il est dans le rôle secondaire. La réplication des données entre les deux noeuds est gérée par DRBD.

Vous pouvez choisir entre la réplication synchrone et asynchrone des données entre les gestionnaires de files d'attente principal et secondaire. Si vous sélectionnez l'option asynchrone, les opérations telles que IBM MQ PUT ou GET se terminent et reviennent à l'application avant que l'événement ne soit répliqué sur le gestionnaire de files d'attente secondaire. Avec la réplication asynchrone, suite à une situation de reprise, il se peut que certaines données de messagerie soient perdues. Toutefois, le gestionnaire de files d'attente secondaire sera cohérent et pourra s'exécuter immédiatement, même s'il est démarré légèrement avant dans le flux de messages.

Vous ne pouvez pas ajouter de reprise après incident à un gestionnaire de files d'attente existant, bien que vous puissiez migrer un gestionnaire de files d'attente existant pour qu'il devienne un gestionnaire de files d'attente de données répliquées (voir «[Migration d'un gestionnaire de files d'attente pour devenir un gestionnaire de files d'attente de données répliquées pour la reprise après incident](#)», à la page 658).

Vous pouvez avoir plusieurs paires de gestionnaires de files d'attente de données répliquées s'exécutant sur un certain nombre de serveurs différents. Par exemple, vous pouvez avoir des gestionnaires de files d'attente de reprise après incident principaux qui s'exécutent sur des noeuds différents, tandis que tous leurs gestionnaires de files d'attente de reprise après incident secondaires s'exécutent sur le même noeud. Des exemples de configuration sont illustrés dans les diagrammes suivants.

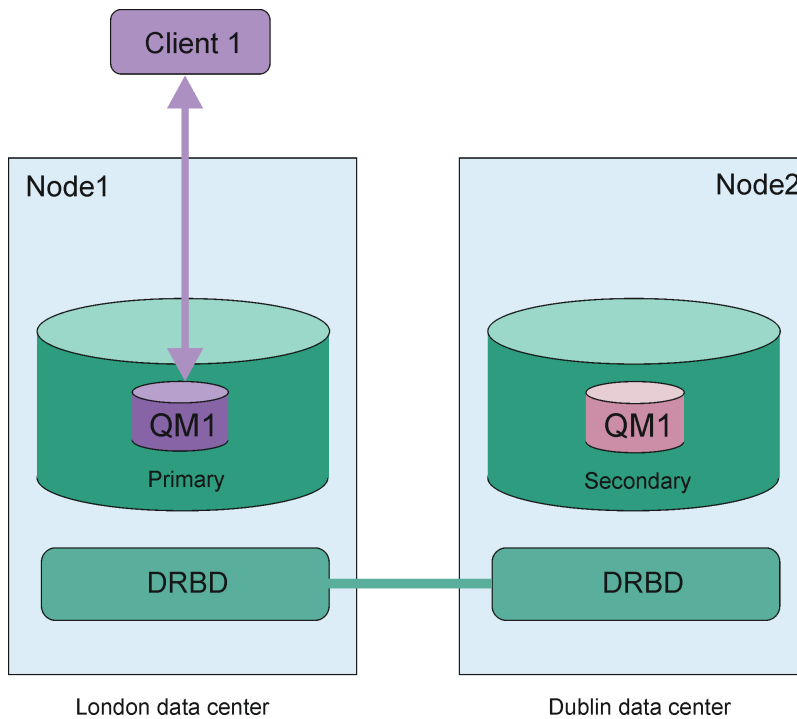


Figure 81. Paire RDQM unique

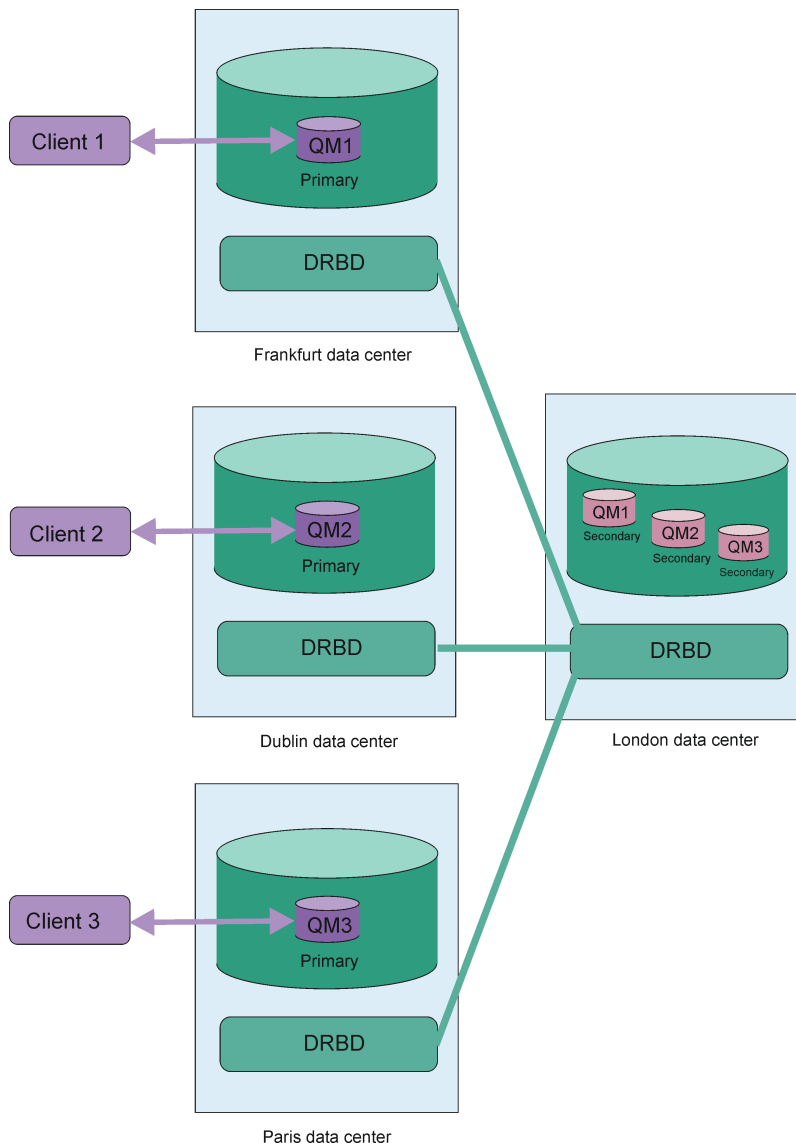


Figure 82. Gestionnaires de files d'attente secondaires dans le même noeud

Réplication, synchronisation et images instantanées

Lorsque les deux noeuds d'une configuration de reprise après incident sont connectés, les mises à jour apportées aux données persistantes d'un gestionnaire de files d'attente de reprise après incident sont transférées de l'instance principale du gestionnaire de files d'attente vers l'instance secondaire. Il s'agit de la **réplication**.

Si la connexion réseau entre les deux noeuds est perdue, les modifications apportées aux données persistantes pour l'instance principale d'un gestionnaire de files d'attente sont suivies. Une fois la connexion réseau restaurée, un processus différent est utilisé pour mettre à niveau l'instance secondaire aussi vite que possible afin qu'elle contienne les mêmes données que l'instance principale. Il s'agit de la **synchronisation**.

Lorsque la synchronisation est en cours, l'état des données sur l'instance secondaire est incohérent. Une **image instantanée** de l'état des données du gestionnaire de files d'attente secondaire est prise. Si une défaillance du noeud principal ou de la connexion réseau se produit lors de la synchronisation, l'instance secondaire revient à cet instantané et le gestionnaire de files d'attente peut être démarré. Cependant, les mises à jour qui ont été effectuées après la défaillance réseau d'origine sont perdues.

Données partitionnées (déconnexion cérébrale)

Les configurations de gestionnaire de files d'attente de données répliquées pour la reprise après incident nécessitent une action de l'utilisateur après la perte de l'instance principale d'un gestionnaire de files d'attente pour promouvoir et exécuter l'instance secondaire sur le noeud de reprise. Il incombe à quiconque (ou à tout autre) promoteur l'instance secondaire de s'assurer que l'ancien gestionnaire de files d'attente principal est arrêté. Si le serveur principal d'origine continue de s'exécuter, il peut traiter des messages et, lorsque le fonctionnement normal est restauré, les deux instances du gestionnaire de files d'attente ont des vues différentes des données. C'est ce que l'on appelle un état de partition ou de division du cerveau.

Vous devez également prendre en compte les éléments suivants :

- Le noeud sur lequel s'exécute le gestionnaire de files d'attente principal échoue complètement. Vous promouvez l'instance secondaire pour qu'elle devienne l'instance principale ; vous ne pouvez pas arrêter l'instance principale d'origine car elle n'est pas en cours d'exécution. Lorsque le noeud d'origine est réparé ou remplacé, le gestionnaire de files d'attente sur ce noeud devient initialement le noeud secondaire et est synchronisé avec le gestionnaire de files d'attente principal sur le noeud de reprise. Les rôles des deux gestionnaires de files d'attente sont alors inversés et le fonctionnement normal recommence. La seule perte de données potentielle dans cette situation est toute donnée que le noeud principal n'a pas encore répliquée sur le noeud secondaire avant l'échec du noeud.
- Un incident réseau affecte le lien de réplication entre les noeuds exécutant les instances principale et secondaire du gestionnaire de files d'attente. Dans ce cas, vous devez vous assurer d'arrêter le serveur principal d'origine avant de promouvoir le serveur secondaire. Si le serveur principal d'origine dispose toujours d'une autre connectivité réseau, deux instances principales s'exécutent en même temps et des données partitionnées peuvent être générées. (Si le lien de réplication fonctionne, vous ne pouvez pas promouvoir un gestionnaire de files d'attente secondaire si l'instance principale est toujours en cours d'exécution, la commande échoue.)
- Un incident réseau complet s'est produit sur le noeud exécutant l'instance principale du gestionnaire de files d'attente. Là encore, vous devez vous assurer d'arrêter l'instance principale avant de promouvoir l'instance secondaire. Si le noeud principal précédent est toujours en cours d'exécution lors de la restauration du réseau, il y aura deux instances principales et les données partitionnées seront à nouveau générées.

Lorsque vous effectuez une reprise en ligne gérée, vous ne devriez pas voir le statut de reprise après incident `partitioned` pour les instances de gestionnaire de files d'attente. Une reprise en ligne gérée arrête le gestionnaire de files d'attente sur le noeud principal, puis démarre le gestionnaire de files d'attente sur le noeud de reprise après la réplication complète des données. Un état partitionné n'est pas attendu car le gestionnaire de files d'attente est arrêté et les données sont synchronisées entre les noeuds avant d'être démarrées sur le noeud de reprise. Si le gestionnaire de files d'attente est démarré sur le noeud de reprise en cas de perte de connectivité entre les noeuds, la divergence des données est probable si le gestionnaire de files d'attente était actif sur le noeud principal en cas de perte de connectivité. Dans ce scénario, un état partitionné doit être signalé une fois la connectivité restaurée car les données du gestionnaire de files d'attente n'ont pas été synchronisées. Si un état partitionné se produit, vous devrez peut-être examiner les deux fichiers et prendre une décision éclairée sur l'ensemble à conserver. Voir [«Résolution d'un problème de partitionnement \(déconnexion cérébrale\) dans un gestionnaire de files d'attente de données répliquées pour la reprise après»](#), à la page 676.

Linux

Conditions requises pour la solution de reprise après incident RDQM

Vous devez répondre à un certain nombre d'exigences avant de configurer une paire de gestionnaires de files d'attente de reprise après incident (DR) de gestionnaire de files d'attente de gestionnaire de files d'attente de données répliquées.

Configuration système requise

Avant de configurer la reprise après incident des gestionnaires de files d'attente de données répliquées, vous devez effectuer une configuration sur chacun des serveurs qui doivent héberger les gestionnaires de files d'attente de reprise après incident des gestionnaires de files d'attente de données répliquées.

- Chaque noeud requiert un groupe de volumes nommé `drbdpool1`. Le stockage de chaque gestionnaire de files d'attente de données répliquées de reprise après incident (DR RDQM) est alloué en tant que deux volumes logiques distincts par gestionnaire de files d'attente à partir de ce groupe de volumes. (Chaque gestionnaire de files d'attente requiert deux volumes logiques pour prendre en charge l'opération de rétablissement de l'image instantanée, de sorte que chaque gestionnaire de files d'attente de données répliquées pour la reprise après incident est alloué un peu plus de deux fois le stockage que vous spécifiez lorsque vous le créez.) Pour des performances optimales, ce groupe de volumes doit être constitué d'un ou de plusieurs volumes physiques correspondant à des unités de disque internes (de préférence des unités SSD).
- Une fois que vous avez créé le groupe de volumes `drbdpool1`, n'y faites rien d'autre. IBM MQ gère les volumes logiques créés dans `drbdpool1`, ainsi que leur mode et leur emplacement de montage.
- Chaque noeud requiert une interface qui est utilisée pour la réplication de données. Cette bande passante doit être suffisante pour prendre en charge les exigences de réplication compte tenu de la charge de travail attendue de tous les gestionnaires de files d'attente de données répliquées.

Pour une tolérance aux pannes maximale, cette interface doit être une carte d'interface réseau (NIC) indépendante.

- DRBD requiert que chaque noeud utilisé pour le gestionnaire de files d'attente de données répliquées ait un nom d'hôte Internet valide (valeur renvoyée par `uname -n`), tel que défini par RFC 952 modifié par RFC 1123.
- S'il existe un pare-feu entre les noeuds utilisés pour les gestionnaires de files d'attente de données répliquées pour la reprise après incident, le pare-feu doit autoriser le trafic entre les noeuds sur les ports utilisés pour la réplication. Un exemple de script est fourni, `/opt/mqm/samp/rdqm/firewalld/configure.sh`, qui ouvre les ports nécessaires si vous exécutez le pare-feu standard dans RHEL. Vous devez exécuter le script en tant que `root`. Si vous utilisez un autre pare-feu, examinez les définitions de service `/usr/lib/firewalld/services/rdqm*` pour voir quels ports doivent être ouverts. Le script ajoute les règles de service `firewalld` permanentes suivantes pour DRBD et IBM MQ (vous pouvez éditer le script pour omettre les ports Pacemaker si vous n'utilisez pas la haute disponibilité):
 - `MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-drbd.xml` autorise les ports TCP 7000-7100.
 - `MQ_INSTALLATION_PATH/samp/rdqm/firewalld/services/rdqm-mq.xml` autorise le port TCP 1414 (vous devez éditer le script si vous avez besoin d'un port différent)
- Si le système utilise SELinux dans un mode autre que permissif, vous devez exécuter la commande suivante:

```
semanage permissive -a drbd_t
```

Configuration requise pour le réseau

Il est recommandé de localiser les noeuds utilisés pour la reprise après incident dans différents centres de données.

Vous devez connaître les limitations suivantes:

- Les performances se dégradent rapidement si le temps d'attente entre les centres de données augmente. IBM prend en charge un temps d'attente pouvant atteindre 5 ms pour la réplication synchrone et 100 ms pour la réplication asynchrone.
- Les données envoyées via le lien de réplication ne font pas l'objet d'un chiffrement supplémentaire au-delà de celui qui pourrait être en place lors de l'utilisation de IBM MQ AMS.
- La configuration d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident entraîne une surcharge en raison de la nécessité de répliquer les données entre les deux noeuds de gestionnaire de files d'attente de données répliquées. La réplication synchrone entraîne une surcharge supérieure à la réplication asynchrone. Lorsque la réplication synchrone est utilisée, les opérations d'E-S de disque sont bloquées jusqu'à ce que les données soient écrites sur les deux noeuds. Lorsque la réplication asynchrone est utilisée, les données doivent uniquement être écrites sur le noeud principal pour que le traitement puisse continuer.

Exigences utilisateur pour l'utilisation des gestionnaires de files d'attente

Pour créer, supprimer ou configurer des gestionnaires de files d'attente de données répliquées (RDQM), vous devez être l'utilisateur root ou disposer d'un ID utilisateur appartenant au groupe mqm qui dispose des droits sudo pour les commandes suivantes:

- **crtmqm**
- **dltmqm**
- **rdqmdz**

Un utilisateur appartenant au groupe mqm peut afficher l'état et le statut d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident à l'aide des commandes suivantes:

- **dspmq**
- **rdqmstatus**

L'utilisateur mqm doit avoir le même UID sur les deux serveurs et le groupe mqm doit avoir le même GID sur les deux serveurs.

Création d'un RDQM de reprise après incident

La commande **crtmqm** permet de créer un gestionnaire de files d'attente de données répliquées (RDQM) qui agit en tant que gestionnaire de files d'attente principal ou secondaire dans une configuration de reprise après incident.

Pourquoi et quand exécuter cette tâche

Vous pouvez créer un gestionnaire de files d'attente de données répliquées (RDQM) en tant qu'utilisateur dans le groupe mqm si l'utilisateur peut utiliser sudo. Sinon, vous devez créer le gestionnaire de files d'attente de données répliquées en tant que racine.

Vous devez créer un gestionnaire de files d'attente de reprise après incident de gestionnaire de files d'attente de données répliquées principal sur un noeud. Vous devez ensuite créer une instance secondaire du même gestionnaire de files d'attente sur un autre noeud. Les instances principale et secondaire doivent avoir le même nom et recevoir la même quantité de stockage.

Les points suivants fournissent des conseils sur le dimensionnement du système de fichiers du gestionnaire de files d'attente:

1. Lors de la création d'un gestionnaire de files d'attente de données répliquées, un système de fichiers est alloué pour stocker les données et les journaux du gestionnaire de files d'attente. Il est important de dimensionner ce système de fichiers de manière appropriée afin que le gestionnaire de files d'attente puisse enregistrer l'activité en cours dans ses journaux et stocker les messages d'application dans les files d'attente. Lors de la définition de la taille du système de fichiers, tenez compte des pics de besoins en matière de messagerie, de la croissance future de la charge de travail et des indisponibilités d'application susceptibles de générer des messages dans les files d'attente. Pour savoir comment calculer la taille du journal de reprise du gestionnaire de files d'attente, voir [«Quelle est la taille de mon système de fichiers journaux?»](#), à la page 715. Lors du calcul des exigences de stockage pour les messages d'application, la taille et le nombre de messages doivent être pris en compte, ainsi que leur en-tête MQMD et les propriétés de message qu'ils possèdent.
2. Les systèmes de fichiers du gestionnaire de files d'attente de données répliquées ne peuvent pas être redimensionnés dynamiquement. Vous devez sauvegarder puis restaurer un gestionnaire de files d'attente de données répliquées (RDQM) avec un système de fichiers plus grand si nécessaire. Voir [«Redimensionnement du système de fichiers pour un gestionnaire de files d'attente de données répliquées à haute disponibilité»](#), à la page 633.
3. Vous pouvez limiter la taille des files d'attente individuelles sur le disque en utilisant des attributs de file d'attente locale, tels que MAXDEPTH et MAXFSIZE. Voir [Modification des fichiers de file d'attente IBM MQ](#).
4. Vous devez surveiller votre utilisation continue du disque et répondre de manière appropriée si l'utilisation du disque augmente avant que l'utilisation du système de fichiers ne devienne critique.

L'utilisation du système de fichiers peut être surveillée à l'aide des fonctions de plateforme / système d'exploitation ou en s'abonnant aux métriques publiées dans les rubriques système IBM MQ décrites dans [Métriques publiées sur les rubriques système](#).

Procédure

- Pour créer un RDQM de reprise après incident principal:

a) Entrez la commande suivante :

```
crtmqm -rr p [-rt (a | s)] -rl Local_IP -ri Recovery_IP -rn Recovery_Name -rp Port  
[other_crtmqm_options] [-fs size] QMname
```

où :

-rr p

Indique que vous créez l'instance principale du gestionnaire de files d'attente.

-rt a | s

-rt s indique que la configuration de reprise après incident utilise la réplication synchrone,

-rt a indique que la configuration de reprise après incident utilise la réplication asynchrone.

La réplication asynchrone est la valeur par défaut.

-rl IP_local

Indique l'adresse IP locale à utiliser pour la réplication de reprise après incident de ce gestionnaire de files d'attente.

-ri IP_reprise

Indique l'adresse IP de l'interface utilisée pour la réplication sur le serveur hébergeant l'instance secondaire du gestionnaire de files d'attente.

-rn nom_reprise

Indique le nom du système qui héberge l'instance secondaire du gestionnaire de files d'attente.

Le nom correspond à la valeur renvoyée si vous exécutez `uname -n` sur ce serveur. Vous devez créer explicitement un gestionnaire de files d'attente secondaire sur ce serveur.

-rp Port

Indique le port à utiliser pour la réplication de reprise après incident.

autre_options_mqm_crtm

Vous pouvez éventuellement spécifier une ou plusieurs des options **crtmqm** générales suivantes:

- -z
- -q
- -c *Texte*
- -d *FileAttenteTransmissionParDéfaut*
- -h *MaxHandles*
- -g *ApplicationGroup*
- -oa *utilisateur|groupe*
- -t *TrigInt*
- -u *DeadQ*
- -x *MaxUMsgs*
- -lp *LogPri*
- -ls *LogSec*
- -lc | -l
- -lla | -lln
- -lf *LogFileTaille*
- -p *port*

-fs taille

Facultatif. Indique la taille du système de fichiers à créer pour le gestionnaire de files d'attente, c'est-à-dire la taille du volume logique créé dans le groupe de volumes drbdpool. Un autre volume logique de cette taille est également créé pour prendre en charge le retour à l'opération d'image instantanée, de sorte que le stockage total pour le gestionnaire de files d'attente de données répliquées pour la reprise après incident est un peu plus de deux fois supérieur à celui spécifié ici.

Taille est une valeur numérique, spécifiée en Go. Vous pouvez indiquer une valeur en Mo en entrant la valeur suivie du caractère M. Par exemple, pour spécifier une taille de système de fichiers de 3 Go, entrez 3. Pour spécifier une taille de système de fichiers de 1024 Mo, entrez 1024M. (Vous pouvez également ajouter un suffixe G pour indiquer explicitement les Go.)

NomGestFile

Indique le nom du gestionnaire de files d'attente de données répliquées. Le nom est dépendant maj/min.

Une fois la commande terminée, elle génère la commande que vous devez entrer sur le noeud secondaire pour créer l'instance secondaire du gestionnaire de files d'attente. Vous pouvez également utiliser la commande **rdqmdr** sur votre noeud principal pour extraire la commande **crtmqm** que vous devez exécuter sur le noeud secondaire pour créer le gestionnaire de files d'attente secondaire. Voir «Gestion des caractéristiques principales et secondaires des gestionnaires de files d'attente de données répliquées pour la reprise après incident», à la page 666.

- Pour créer un RDQM de reprise après incident secondaire:
 - a) Entrez la commande suivante sur le noeud qui doit héberger les instances secondaires du gestionnaire de files d'attente de données répliquées:

```
crtmqm -rr s [-rt (a | s)] -rl Local_IP -ri Primary_IP -rn Primary_Name -rp Port  
[other_crtmqm_options] [-fs size] QMname
```

où :

-rr s

Indique que vous créez l'instance secondaire du gestionnaire de files d'attente.

-rt a | s

-rt s indique que la configuration de reprise après incident utilise la réplication synchrone,
-rt a indique que la configuration de reprise après incident utilise la réplication asynchrone.

-rl IP_local

Indique l'adresse IP locale à utiliser pour la réplication de reprise après incident de ce gestionnaire de files d'attente.

-ri IP_principale

Indique l'adresse IP de l'interface utilisée pour la réplication sur le serveur hébergeant l'instance principale du gestionnaire de files d'attente.

-rn Nom_principale

Indique le nom du système qui héberge l'instance principale du gestionnaire de files d'attente. Le nom correspond à la valeur renvoyée si vous exécutez `uname -n` sur ce serveur.

-rp Port

Indique le port à utiliser pour la réplication de reprise après incident.

autre_options_mqm_crtm

Vous pouvez éventuellement spécifier une ou plusieurs des options **crtmqm** générales suivantes:

- -z

-fs taille

Indique la taille du système de fichiers à créer pour le gestionnaire de files d'attente, c'est-à-dire la taille du volume logique créé dans le groupe de volumes drbdpool. Si vous avez spécifié

une taille autre que la taille par défaut lors de la création du gestionnaire de files d'attente principal, vous devez spécifier la même valeur ici.

Taille est une valeur numérique, spécifiée en Go. Vous pouvez indiquer une valeur en Mo en entrant la valeur suivie du caractère M. Par exemple, pour spécifier une taille de système de fichiers de 3 Go, entrez 3. Pour spécifier une taille de système de fichiers de 1024 Mo, entrez 1024M. (Vous pouvez également ajouter un suffixe G pour indiquer explicitement les Go.)

NomGestFile

Indique le nom du gestionnaire de files d'attente de données répliquées. Ce nom doit être identique à celui que vous avez spécifié pour l'instance principale du gestionnaire de files d'attente. Notez que le nom est sensible à la casse.

Que faire ensuite

Une fois que vous avez créé vos instances principale et secondaire de votre gestionnaire de files d'attente, vous devez vérifier le statut sur les deux noeuds pour vous assurer que les deux sont corrects. Utilisez la commande **rdqmstatus** sur les deux noeuds. Les noeuds doivent afficher le statut normal, comme décrit dans «Affichage du statut des gestionnaires de files d'attente de données», à la page 667. S'ils n'affichent pas ce statut, supprimez l'instance secondaire et recréez-la en prenant soin d'utiliser les arguments corrects.

Référence associée

[crtmqm](#)

Suppression d'un RDQM de reprise après incident

La commande **dltmqm** permet de supprimer un gestionnaire de files d'attente de données répliquées de reprise après incident (RDQM).

Pourquoi et quand exécuter cette tâche

Vous devez exécuter la commande pour supprimer le gestionnaire de files d'attente de données répliquées sur les noeuds principal et secondaire du gestionnaire de files d'attente de données répliquées. Le gestionnaire de files d'attente de données répliquées doit d'abord être arrêté. Vous pouvez exécuter la commande en tant qu'utilisateur mqm si cet utilisateur dispose des privilèges sudo nécessaires. Sinon, vous devez exécuter la commande en tant que root.

Procédure

- Pour supprimer un gestionnaire de files d'attente de données répliquées pour la reprise après incident, entrez la commande suivante:

```
dltmqm RDQM_name
```

Référence associée

[dltmqm](#)

Migration d'un gestionnaire de files d'attente pour devenir un gestionnaire de files d'attente de données répliquées pour la reprise après incident

Vous pouvez migrer un gestionnaire de files d'attente existant pour qu'il devienne un gestionnaire de files d'attente de données répliquées (RDQM) de reprise après incident en sauvegardant ses données persistantes, puis en restaurant les données dans un gestionnaire de files d'attente de gestionnaires de files d'attente de données répliquées nouvellement créé portant le même nom.

Pourquoi et quand exécuter cette tâche

Les gestionnaires de files d'attente de données répliquées pour la reprise après incident requièrent un volume logique dédié (système de fichiers) et la configuration de la répllication de disque. Ces composants sont configurés uniquement lorsqu'un nouveau gestionnaire de files d'attente est créé. Un gestionnaire de files d'attente existant peut être migré pour utiliser un gestionnaire de files d'attente de

données répliquées en sauvegardant ses données persistantes, puis en restaurant les données dans un gestionnaire de files d'attente de données répliquées nouvellement créé et portant le même nom. Cette procédure permet de conserver la configuration, l'état et les messages persistants du gestionnaire de files d'attente au moment de la création de la sauvegarde.

Remarque : Vous ne pouvez migrer un gestionnaire de files d'attente qu'à partir d'une version d' IBM MQ identique ou antérieure à la version sur laquelle le gestionnaire de files d'attente de données répliquées est installé. Le système d'exploitation et l'architecture doivent également être identiques. Sinon, vous devez créer un gestionnaire de files d'attente sur votre plateforme cible. Voir [Déplacement d'un gestionnaire de files d'attente vers un autre système d'exploitation](#).

Vous devez remplir les conditions suivantes avant de migrer un gestionnaire de files d'attente:

- Évaluez vos exigences en matière de reprise après incident et voir [«Reprise après incident RDQM»](#), à la page 650.
- Passez en revue les applications et les gestionnaires de files d'attente qui se connectent au gestionnaire de files d'attente. Prenez en compte les modifications requises pour router les connexions vers le noeud RDQM sur lequel le gestionnaire de files d'attente est en cours d'exécution.
- Mettez à disposition ou identifiez des noeuds de gestionnaire de files d'attente de données répliquées existants pour la configuration choisie. Pour plus d'informations sur la configuration système requise pour les gestionnaires de files d'attente de données répliquées, voir [«Conditions requises pour la solution de reprise après incident RDQM»](#), à la page 653.
- Installez IBM MQ Advanced, qui inclut la fonction RDQM, sur chaque noeud.
- Vous pouvez éventuellement vérifier la configuration du gestionnaire de files d'attente de données répliquées à l'aide d'un gestionnaire de files d'attente de test, qui peut ensuite être supprimé. Il est recommandé de tester la configuration afin d'identifier et de résoudre les problèmes avant de migrer le gestionnaire de files d'attente.
- Vérifiez la configuration de sécurité du gestionnaire de files d'attente, puis répliquez les utilisateurs et les groupes locaux requis sur chaque noeud de gestionnaire de files d'attente de données répliquées.
- Vérifiez la configuration du gestionnaire de files d'attente et du canal pour déterminer si des exits d'API, des exits de canal ou des exits de conversion de données sont utilisés. Installez les exits requis sur chaque noeud RDQM.
- Passez en revue les services de gestionnaire de files d'attente qui ont été définis, puis installez et configurez les processus requis sur chaque noeud RDQM.

Procédure

1. Sauvegardez le gestionnaire de files d'attente existant:

- a) Arrêtez le gestionnaire de files d'attente existant en émettant une commande d'arrêt d'attente `endmqm -wou` une commande d'arrêt immédiat `endmqm -i`. Cette étape est importante pour garantir la cohérence des données de la sauvegarde.
- b) Déterminez l'emplacement du répertoire de données du gestionnaire de files d'attente en affichant le fichier de configuration IBM MQ , `mqmqs.ini`. Sous Linux, ce fichier se trouve dans le répertoire `/var/mqm`. Pour plus d'informations sur `mqmqs.ini`, voir [«Fichier de configuration IBM MQ , mqmqs.ini»](#), à la page 98.

Recherchez la section `QueueManager` du gestionnaire de files d'attente dans le fichier. Si la strophe contient une clé nommée `DataPath`, sa valeur est le répertoire de données du gestionnaire de files d'attente. Si la clé n'existe pas, le répertoire de données du gestionnaire de files d'attente peut être déterminé à l'aide des valeurs des clés `Prefix` et `Directory`. Le répertoire de données du gestionnaire de files d'attente est une concaténation de ces valeurs, au format `préfixe/qmgrs/répertoire`. Pour plus d'informations sur la section `QueueManager`, voir [«Section QueueManager du fichier mqmqs.ini»](#), à la page 109.

- c) Créez une sauvegarde du répertoire de données du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande `tar`. Par exemple, pour sauvegarder

le répertoire de données d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (point) unique:

```
tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
```

- d) Déterminez l'emplacement du répertoire de journaux du gestionnaire de files d'attente en affichant le IBM MQ fichier de configuration du gestionnaire de files d'attente `qm.ini`. Ce fichier se trouve dans le répertoire de données du gestionnaire de files d'attente. Pour plus d'informations sur le fichier, voir [«Fichiers de configuration du gestionnaire de files d'attente, qm.ini»](#), à la page 111.

Le répertoire de journaux du gestionnaire de files d'attente est défini en tant que valeur de la clé `LogPath` dans la section `Log`. Pour plus d'informations sur la strophe, voir [«Section de journal du fichier qm.ini»](#), à la page 147.

- e) Créez une sauvegarde du répertoire de journaux du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande `tar`. Par exemple, pour sauvegarder le répertoire de journaux d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (point) unique:

```
tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
```

- f) Créez une sauvegarde des référentiels de certificats utilisés par le gestionnaire de files d'attente s'ils ne se trouvent pas dans le répertoire de données du gestionnaire de files d'attente. Assurez-vous que le fichier de la base de données de clés et le fichier de mot de passe secret sont sauvegardés. Pour plus d'informations sur le référentiel de clés du gestionnaire de files d'attente, voir [Référentiel de clés SSL/TLS et Localisation du référentiel de clés pour un gestionnaire de files d'attente](#). Pour plus d'informations sur la localisation du magasin de clés AMS si le gestionnaire de files d'attente est configuré pour utiliser l'interception MCA (Message Channel Agent) AMS, voir [Interception MCA \(Message Channel Agent\)](#).

- g) Le gestionnaire de files d'attente existant n'est plus nécessaire et peut donc être supprimé. Toutefois, dans la mesure du possible, vous ne devez supprimer le gestionnaire de files d'attente existant qu'une fois qu'il a été correctement restauré sur le système cible. Le report de la suppression garantit que le gestionnaire de files d'attente peut être redémarré si le processus de migration n'aboutit pas.

Remarque : Si vous reportez la suppression du gestionnaire de files d'attente existant, ne le redémarrez pas. Il est important que le gestionnaire de files d'attente reste arrêté car d'autres modifications apportées à sa configuration ou à son état sont perdues lors de la migration.

2. Préparez le noeud RDQM principal:

- a) Créez un gestionnaire de files d'attente de données répliquées (RDQM) portant le même nom que le gestionnaire de files d'attente que vous avez sauvegardé. Vérifiez que le système de fichiers alloué au gestionnaire de files d'attente de données répliquées par `crtmqm` est suffisamment grand pour contenir les données, les journaux principaux et les journaux secondaires du gestionnaire de files d'attente existant, ainsi qu'un espace supplémentaire pour une extension ultérieure. Pour plus d'informations sur la création d'un gestionnaire de files d'attente de données répliquées, voir [«Création d'un RDQM de reprise après incident»](#), à la page 655.
- b) Déterminez le noeud de gestionnaire de files d'attente de données répliquées principal pour le gestionnaire de files d'attente. Pour plus d'informations sur la détermination du noeud principal, voir `rdqmstatus` ([display RDQM status](#)).
- c) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, si le gestionnaire de files d'attente de données répliquées est démarré, arrêtez-le à l'aide de la commande `endmqm -w` ou `endmqm -i`.
- d) Déterminez l'emplacement des répertoires de données et de journaux pour le gestionnaire de files d'attente de gestionnaires de files d'attente de données répliquées (utilisez les méthodes décrites dans les étapes 1b et 1d).
- e) Supprimez le contenu des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées, mais pas les répertoires eux-mêmes.

3. Restaurez le gestionnaire de files d'attente sur le noeud RDQM principal:

- a) Copiez les sauvegardes des données et des répertoires de journaux du gestionnaire de files d'attente sur le noeud RDQM principal, ainsi que les sauvegardes distinctes des référentiels de certificats utilisés par le gestionnaire de files d'attente.
- b) Restaurez la sauvegarde du répertoire de données du gestionnaire de files d'attente dans le répertoire de données vide du nouveau gestionnaire de files d'attente de données répliquées, en veillant à préserver la propriété et les droits d'accès aux fichiers. Si la sauvegarde a été créée à l'aide de l'exemple de commande tar à l'étape 1c , la commande suivante peut être utilisée par le superutilisateur pour la restaurer:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

- c) Restaurez la sauvegarde du répertoire de journaux du gestionnaire de files d'attente dans le répertoire de journaux vide du nouveau gestionnaire de files d'attente de données répliquées, en veillant à préserver la propriété et les droits d'accès aux fichiers. Si la sauvegarde a été créée à l'aide de l'exemple de commande tar de l'étape 1e , la commande suivante peut être utilisée par le superutilisateur pour la restaurer:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

- d) Editez le fichier de configuration de gestionnaire de files d'attente restauré, `qm.ini`, dans le répertoire de données du gestionnaire de files d'attente de données répliquées. Mettez à jour la valeur de la clé `LogPath` dans la section `Log` pour spécifier le répertoire de journaux du gestionnaire de files d'attente de données répliquées (RDQM).

Passez en revue les autres chemins de fichier définis dans le fichier de configuration et mettez-les à jour si nécessaire. Par exemple, vous devrez peut-être mettre à jour les chemins suivants:

- Chemin d'accès aux fichiers journaux d'erreurs générés par les services de message de diagnostic.
- Chemin d'accès aux exits requis par le gestionnaire de files d'attente.
- Chemin d'accès aux fichiers de commutation de chargement si le gestionnaire de files d'attente est un coordinateur de transaction XA.

- e) Si le gestionnaire de files d'attente est configuré pour utiliser l'interception AMS Message Channel Agent (MCA), copiez le magasin de clés AMS dans la nouvelle installation du gestionnaire de files d'attente de données répliquées, puis vérifiez et mettez à jour la configuration. Le magasin de clés doit être disponible sur chaque noeud de gestionnaire de files d'attente de données répliquées. Par conséquent, s'il ne se trouve pas dans le système de fichiers répliqué pour le gestionnaire de files d'attente, il doit être copié sur chaque noeud. Pour plus d'informations, voir [Interception de l'agent MCA \(Message Channel Agent\)](#).

- f) Vérifiez que le gestionnaire de files d'attente est affiché par la commande `dspmq` et que son statut est signalé comme étant arrêté. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente de données répliquées (RDQM DR):

```
$ dspmq -o status -o dr
QMNAME(QM1) STATUS(Ended normally) DRROLE(Primary)
```

- g) Vérifiez que les données du gestionnaire de files d'attente restaurées ont été répliquées sur les noeuds de gestionnaire de files d'attente de données répliquées secondaires à l'aide de la commande `rdqmstatus` pour afficher le statut du gestionnaire de files d'attente. Le statut de la reprise après incident doit être signalé comme `Normal` sur chaque noeud. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente de données répliquées (RDQM DR):

```
$ rdqmstatus -m QM1
Queue manager status:           Ended normally
Queue manager file system:      51MB used, 1.0GB allocated [5%]
DR role:                        Primary
DR status:                      Normal
DR type:                        Synchronous
DR port:                        3000
```

DR local IP address:	192.168.20.1
DR remote IP address:	192.168.20.2

- h) Démarrez le gestionnaire de files d'attente sur le noeud RDQM principal.
- i) Connectez-vous au gestionnaire de files d'attente et mettez à jour la valeur de l'attribut de gestionnaire de files d'attente SSLKEYR pour spécifier le nouvel emplacement du référentiel de certificats du gestionnaire de files d'attente. Par défaut, la valeur de cet attribut est `queue_manager_data_directory/ssl/key`. Le référentiel de certificats doit se trouver au même emplacement sur chaque noeud RDQM. Si le référentiel ne se trouve pas dans le système de fichiers répliqué pour le gestionnaire de files d'attente, il doit être copié sur chaque noeud à la place.
- j) Passez en revue les définitions d'objet IBM MQ pour le gestionnaire de files d'attente et mettez à jour la valeur des attributs d'objet qui font référence aux paramètres réseau modifiés, au répertoire d'installation de IBM MQ ou au répertoire de données du gestionnaire de files d'attente, y compris les objets suivants:
- Adresses IP locales utilisées par les programmes d'écoute (attributIPADDR).
 - Adresses IP locales utilisées par les canaux (attributLOCLADDR).
 - Adresses IP locales définies pour les canaux récepteurs de cluster (attributCONNAME).
 - Adresses IP locales définies pour les objets d'informations de communication (attributGRPADDR).
 - Chemins système définis pour les définitions d'objet de processus et de service.
- k) Arrêtez et redémarrez le gestionnaire de files d'attente pour vous assurer que les modifications prennent effet.
- l) Répétez l'étape 3j pour les gestionnaires de files d'attente éloignées, ainsi que les paramètres équivalents pour les applications qui se connectent au gestionnaire de files d'attente migré, notamment:
- Noms de connexion de canal (attributCONNAME).
 - Règles d'authentification de canal qui restreignent les connexions entrantes du gestionnaire de files d'attente en fonction de son adresse IP ou de son nom d'hôte.
 - Tables de définition de canal du client (CCDT), paramètres de nom de domaine (DNS), routage réseau ou informations de connexion équivalentes.
- m) Effectuez une reprise en ligne gérée du gestionnaire de files d'attente sur chaque noeud de gestionnaire de files d'attente de données répliquées pour vous assurer que la configuration requise a été correctement établie. Voir [«Basculer vers un noeud de reprise»](#), à la page 672.

Redimensionnement du système de fichiers pour un gestionnaire de files d'attente de données répliquées pour la reprise après incident

Pour redimensionner le système de fichiers d'un gestionnaire de files d'attente de données répliquées (RDQM) de reprise après incident existant, sauvegardez ses données persistantes, puis restaurez les données dans un gestionnaire de files d'attente de gestionnaires de files d'attente de données répliquées nouvellement créé ayant le même nom mais un système de fichiers de taille différente.

Pourquoi et quand exécuter cette tâche

Les gestionnaires de files d'attente de données répliquées pour la reprise après incident requièrent un volume logique dédié (système de fichiers) et la configuration de la réplication de disque. Ces composants sont configurés uniquement lorsqu'un nouveau gestionnaire de files d'attente est créé. Le système de fichiers ne peut pas être redimensionné une fois qu'il a été créé car il doit avoir la même taille sur chaque noeud. Pour redimensionner le système de fichiers d'un gestionnaire de files d'attente de données répliquées (RDQM) existant, vous pouvez sauvegarder ses données persistantes, puis restaurer les données dans un gestionnaire de files d'attente de données répliquées nouvellement créé qui a le même nom mais un système de fichiers d'une taille différente. Cette procédure permet de conserver la configuration, l'état et les messages persistants du gestionnaire de files d'attente au moment de la création de la sauvegarde.

Procédure

1. Sauvegardez le gestionnaire de files d'attente de données répliquées existant sur le noeud de gestionnaire de files d'attente de données répliquées principal:

- a) Déterminez le noeud de gestionnaire de files d'attente de données répliquées principal pour le gestionnaire de files d'attente. Pour plus d'informations sur la détermination du noeud principal, voir [rdqmstatus \(display RDQM status\)](#) .
- b) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, si le gestionnaire de files d'attente de données répliquées est démarré, arrêtez-le à l'aide de la commande **endmqm -w** ou **endmqm -i** .
- c) Déterminez l'emplacement du répertoire de données du gestionnaire de files d'attente en affichant le fichier de configuration IBM MQ , `mqs.ini`. Sous Linux, ce fichier se trouve dans le répertoire `/var/mqm` . Pour plus d'informations sur `mqs.ini`, voir «[Fichier de configuration IBM MQ , mqs.ini](#)», à la page 98.

Recherchez la section `QueueManager` du gestionnaire de files d'attente dans le fichier. Le répertoire de données du gestionnaire de files d'attente est la valeur de la clé nommée `DataPath`. Pour plus d'informations sur la section `QueueManager` , voir «[Section QueueManager du fichier mqs.ini](#)», à la page 109.

- d) Créez une sauvegarde du répertoire de données du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande **tar** . Par exemple, pour sauvegarder le répertoire de données d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (.) unique:

```
tar -cvzf qm-data.tar.gz -C queue_manager_data_dir .
```

- e) Déterminez l'emplacement du répertoire de journaux du gestionnaire de files d'attente en affichant le IBM MQ fichier de configuration du gestionnaire de files d'attente `qm.ini`. Ce fichier se trouve dans le répertoire de données du gestionnaire de files d'attente. Pour plus d'informations sur le fichier, voir «[Fichiers de configuration du gestionnaire de files d'attente, qm.ini](#)», à la page 111.

Le répertoire de journaux du gestionnaire de files d'attente est défini en tant que valeur de la clé `LogPath` dans la section `Journal`. Pour plus d'informations sur la strophe, voir «[Section de journal du fichier qm.ini](#)», à la page 147.

- f) Créez une sauvegarde du répertoire de journaux du gestionnaire de files d'attente. Sous Linux, vous pouvez effectuer cette opération à l'aide de la commande **tar** . Par exemple, pour sauvegarder le répertoire de journaux d'un gestionnaire de files d'attente, vous pouvez utiliser la commande suivante. Notez le dernier paramètre de la commande, qui est un point (.) unique:

```
tar -cvzf qm-log.tar.gz -C queue_manager_log_dir .
```

- g) Supprimez le gestionnaire de files d'attente de données répliquées existant.

2. Restaurez le gestionnaire de files d'attente avec un système de fichiers de la taille requise:

- a) Créez un gestionnaire de files d'attente de données répliquées (RDQM) portant le même nom que le gestionnaire de files d'attente que vous avez sauvegardé. Vérifiez que le système de fichiers alloué au gestionnaire de files d'attente de données répliquées par **crtmqm** correspond à la taille requise et qu'il est suffisamment grand pour contenir les données, les journaux principaux et les journaux secondaires du gestionnaire de files d'attente existant, plus un espace supplémentaire pour une extension ultérieure. Pour plus d'informations sur la création d'un gestionnaire de files d'attente de données répliquées, voir «[Création d'un RDQM de reprise après incident](#)», à la page 655.
- b) Déterminez le noeud de gestionnaire de files d'attente de données répliquées principal pour le gestionnaire de files d'attente. Pour plus d'informations sur la détermination du noeud principal, voir [rdqmstatus \(display RDQM status\)](#).
- c) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, si le gestionnaire de files d'attente de données répliquées est démarré, arrêtez-le à l'aide de la commande **endmqm -w** ou **endmqm -i** .

- d) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, déterminez le nouvel emplacement des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées (utilisez les méthodes décrites dans les étapes 1c et 1e).
- e) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, supprimez le contenu des répertoires de données et de journaux du gestionnaire de files d'attente de données répliquées, mais pas les répertoires eux-mêmes.
- f) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, restaurez la sauvegarde du répertoire de données du gestionnaire de files d'attente dans le répertoire de données vide du nouveau gestionnaire de files d'attente de données répliquées, en vous assurant que la propriété des fichiers et les droits d'accès sont préservés. Si la sauvegarde a été créée à l'aide de l'exemple de commande **tar** à l'étape 1d, la commande suivante peut être utilisée par le superutilisateur pour la restaurer:

```
tar -xvzpf qm-data.tar.gz -C queue_manager_data_dir
```

- g) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, restaurez la sauvegarde du répertoire de journaux du gestionnaire de files d'attente dans le répertoire de journaux vide du nouveau gestionnaire de files d'attente de données répliquées, en veillant à préserver la propriété et les droits d'accès aux fichiers. Si la sauvegarde a été créée à l'aide de l'exemple de commande **tar** à l'étape 1f, la commande suivante peut être utilisée par l'utilisateur root pour la restaurer:

```
tar -xvzpf qm-log.tar.gz -C queue_manager_log_dir
```

- h) Sur le noeud de gestionnaire de files d'attente de données répliquées principal, éditez le fichier de configuration de gestionnaire de files d'attente restauré, `qm.ini`, dans le répertoire de données du nouveau gestionnaire de files d'attente de données répliquées. Mettez à jour la valeur de la clé `LogPath` dans la section `Log` afin de spécifier le répertoire de journaux pour le nouveau gestionnaire de files d'attente de données répliquées que vous avez déterminé à l'étape 2d. Passez en revue les autres chemins de fichier définis dans le fichier de configuration et mettez-les à jour si nécessaire. Par exemple, vous devrez peut-être mettre à jour les chemins suivants:
- Chemin d'accès aux fichiers journaux d'erreurs générés par les services de message de diagnostic.
 - Chemin d'accès aux exits requis par le gestionnaire de files d'attente.
 - Chemin d'accès aux fichiers de commutation de chargement si le gestionnaire de files d'attente est un coordinateur de transaction XA.
- i) Vérifiez que le gestionnaire de files d'attente est affiché par la commande **dspm** et que son statut est ended. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente de données répliquées (RDQM DR):

```
$ dspm -o status -o dr
QMNAME(QM1) STATUS(Ended normally) DR(Primary)
```

- j) Vérifiez que les données du gestionnaire de files d'attente restaurées ont été répliquées sur le noeud de gestionnaire de files d'attente de données répliquées secondaire à l'aide de la commande **rdqmstatus** pour afficher le statut du gestionnaire de files d'attente. Le statut de la reprise après incident doit être signalé comme `Normal` sur chaque noeud. L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente de reprise après incident de gestionnaire de files d'attente de données répliquées sur le noeud principal:

```
$ rdqmstatus -m QM1
Queue manager status:      Running
CPU:                       0.00
Memory:                    123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
DR role:                   Primary
DR status:                 Normal
DR type:                   Synchronous
DR port:                   3000
```



```
DR local IP address: 192.168.20.1
DR remote IP address: 192.168.20.2
```

L'exemple suivant illustre un exemple de sortie pour un gestionnaire de files d'attente de données répliquées pour la reprise après incident sur le noeud de reprise:

```
Queue manager status: Ended immediately
DR role: Secondary
DR status: Normal
DR port: 3000
DR local IP address: 192.168.20.2
DR remote IP address: 192.168.20.1
```

- k) Démarrez le gestionnaire de files d'attente sur le noeud RDQM principal.
- l) Effectuez une commutation du gestionnaire de files d'attente sur le noeud de reprise pour vous assurer que la configuration requise a été correctement établie. Voir [«Basculement vers un noeud de reprise»](#), à la page 672.

Stockage du statut de l'application persistante

Vous pouvez stocker des informations de statut persistantes relatives aux applications ainsi que d'autres données de gestionnaire de files d'attente.

Chaque gestionnaire de files d'attente IBM MQ dispose d'un système de fichiers dédié pour son état persistant, qui inclut à la fois ses données de file d'attente et son journal de reprise. Dans une configuration de gestionnaire de files d'attente de données répliquées, le système de fichiers est sauvegardé par un volume logique qui est répliqué entre les systèmes Linux (noeuds). Le système de fichiers inclut un répertoire `userdata` que vous pouvez utiliser pour stocker les informations de statut persistantes de vos applications. Par conséquent, lorsqu'un gestionnaire de files d'attente de données répliquées se déplace pour s'exécuter sur un autre noeud de votre configuration de gestionnaire de files d'attente de données répliquées, vous disposez d'un contexte d'application et d'un contexte de gestionnaire de files d'attente. Voir [Contenu d'annuaire sur les systèmes Unix et Linux](#).

Si vous choisissez de stocker l'état de l'application dans le répertoire `userdata`, vous devez savoir que les données écrites à cet emplacement peuvent consommer l'espace disque disponible alloué au gestionnaire de files d'attente. Vous devez vous assurer que l'espace disque disponible reste suffisant pour que le gestionnaire de files d'attente puisse écrire des données de file d'attente, des journaux et d'autres informations d'état persistantes.

Le répertoire `userdata` est propriétaire de l'utilisateur et du groupe `mqm` et est accessible en lecture à tous, de sorte que les utilisateurs puissent y accéder sans avoir à faire partie du groupe d'administrateurs IBM MQ (c'est-à-dire `mqm`). Vous ne pouvez pas modifier les droits du répertoire `userdata`, mais vous pouvez y créer du contenu avec la propriété et les droits dont vous avez besoin.

Lors d'une reprise en ligne du gestionnaire de files d'attente de données répliquées, le gestionnaire de files d'attente est arrêté et son système de fichiers est démonté sur son noeud de gestionnaire de files d'attente de données répliquées en cours. Le système de fichiers est ensuite monté et le gestionnaire de files d'attente est redémarré sur un autre noeud dans la configuration du gestionnaire de files d'attente de données répliquées. Un système de fichiers ne peut pas être démonté si un processus a un descripteur ouvert pour l'un de ses fichiers. Pour s'assurer que la reprise en ligne d'un gestionnaire de files d'attente peut être terminée, si le système de fichiers du gestionnaire de files d'attente ne peut pas être démonté, les processus ayant un descripteur de fichier ouvert reçoivent un signal `SIGTERM`, suivi d'un signal `SIGKILL` si les descripteurs ouverts ne sont pas libérés. Vos applications doivent être conçues pour répondre correctement à `SIGTERM`. Si des applications ou des processus sont configurés en tant que service de gestionnaire de files d'attente, lors d'une reprise en ligne gérée, ils peuvent être arrêtés lors de l'arrêt du gestionnaire de files d'attente avant que le système de fichiers ne soit démonté. Si une application ou un processus n'est pas configuré en tant que service de gestionnaire de files d'attente ou qu'une reprise en ligne non gérée se produit, telle qu'une perte de quorum, il est probable que des signaux seront envoyés pour libérer le système de fichiers.

Gestion des caractéristiques principales et secondaires des gestionnaires de files d'attente de données répliquées pour la reprise après incident

Vous pouvez changer un gestionnaire de files d'attente de données répliquées de reprise après incident secondaire (DR RDQM) en gestionnaire de files d'attente de données répliquées de reprise après incident principal. Vous pouvez également changer une instance principale en instance secondaire.

Pourquoi et quand exécuter cette tâche

Vous utilisez la commande **rdqmdr** pour changer une instance secondaire d'un gestionnaire de files d'attente de données répliquées en instance principale. Vous devrez peut-être effectuer cette action si vous perdez votre instance principale pour une raison quelconque. Vous pouvez ensuite démarrer le gestionnaire de files d'attente et continuer à l'exécuter sur le noeud de reprise.

Vous pouvez également utiliser la commande **rdqmdr** pour changer une instance principale d'un gestionnaire de files d'attente de données répliquées en instance secondaire. Vous devrez peut-être effectuer cette action, par exemple, si vous reconfigurez votre système.

Vous pouvez également utiliser **rdqmdr** sur un gestionnaire de files d'attente principal pour extraire la commande exacte dont vous avez besoin pour créer une instance secondaire de ce gestionnaire de files d'attente sur votre noeud de reprise.

Vous pouvez utiliser la commande **rdqmdr** en tant qu'utilisateur du groupe mqm si l'utilisateur peut utiliser sudo. Sinon, vous devez être connecté en tant que superutilisateur.

Procédure

- Pour changer une instance secondaire d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident en instance principale, entrez la commande suivante:

```
rdqmdr -m QMname -p
```

Cette commande échoue si l'instance principale du gestionnaire de files d'attente est toujours en cours d'exécution et que le lien de répllication de reprise après incident fonctionne toujours.

- Pour changer une instance principale du gestionnaire de files d'attente en instance secondaire, entrez la commande suivante:

```
rdqmdr -m QMname -s
```

- Pour afficher la commande **crtmqm** requise pour configurer l'instance secondaire d'un gestionnaire de files d'attente, entrez la commande suivante sur votre noeud principal:

```
rdqmdr -d -m QMname
```

Vous pouvez entrer la commande **crtmqm** renvoyée sur votre noeud secondaire pour créer l'instance secondaire du gestionnaire de files d'attente de données répliquées (RD RDQM).

Démarrage, arrêt et affichage de l'état d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident

Vous utilisez des variantes des commandes de contrôle IBM MQ standard pour démarrer, arrêter et afficher l'état en cours d'un gestionnaire de files d'attente de données répliquées de reprise après incident (DR RDQM).

Pourquoi et quand exécuter cette tâche

Vous devez exécuter les commandes qui démarrent, arrêtent et affichent l'état en cours d'un gestionnaire de files d'attente de données répliquées (RDQM) en tant qu'utilisateur appartenant au groupe mqm .

Vous devez exécuter les commandes pour démarrer et arrêter un gestionnaire de files d'attente sur le noeud principal de ce gestionnaire de files d'attente (c'est-à-dire le noeud sur lequel le gestionnaire de files d'attente est en cours d'exécution).

Procédure

- Pour démarrer un gestionnaire de files d'attente de données répliquées pour la reprise après incident, entrez la commande suivante sur le noeud principal du gestionnaire de files d'attente de données répliquées:

```
strmqm qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées que vous souhaitez démarrer.

- Pour arrêter un gestionnaire de files d'attente de données répliquées, entrez la commande suivante sur le noeud principal du gestionnaire de files d'attente de données répliquées:

```
endmqm qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées que vous souhaitez arrêter.

- Pour afficher l'état d'un gestionnaire de files d'attente de données répliquées, entrez la commande suivante:

```
dspmqr -m QMname
```

Les informations d'état qui sont générées varient selon que vous exécutez la commande sur le noeud principal ou secondaire du gestionnaire de files d'attente de données répliquées. S'il est exécuté sur le noeud principal, l'un des messages de statut normal renvoyés par **dspmqr** s'affiche. Si vous exécutez la commande sur un noeud secondaire, l'état Ended `immediately` s'affiche. Par exemple, si **dspmqr** est exécuté sur le noeud RDQM7, les informations suivantes peuvent être renvoyées:

```
QMNAME(DRQM8)          STATUS(Ended immediately)
QMNAME(DRQM7)          STATUS(Running)
```

Vous pouvez utiliser des arguments avec `dspmqr` pour déterminer si un gestionnaire de files d'attente de données répliquées est configuré pour la reprise après incident et s'il s'agit actuellement de l'instance principale ou secondaire:

```
dspmqr -m QMname -o (dr | DR)
```

L'une des réponses suivantes s'affiche:

DRROLE()

Indique que le gestionnaire de files d'attente n'est pas configuré pour la reprise après incident.

DRROLE(Primary)

Indique que le gestionnaire de files d'attente est configuré en tant que serveur principal de reprise après incident.

DRROLE(Secondary)

Indique que le gestionnaire de files d'attente est configuré en tant que serveur secondaire de reprise après incident.

Référence associée

[dspmqr](#)

[endmqm](#)

[strmqm](#)

Linux

Affichage du statut des gestionnaires de files d'attente de données

Vous pouvez afficher le statut de tous les gestionnaires de files d'attente de données répliquées de reprise après incident (DR RDQM) sur un noeud ou des informations détaillées sur un gestionnaire de files d'attente de données répliquées de reprise après incident spécifié.

Pourquoi et quand exécuter cette tâche

La commande **rdqmstatus** permet d'afficher le statut de tous les gestionnaires de files d'attente de données répliquées pour la reprise après incident ou des gestionnaires de files d'attente de données répliquées individuels.

Le statut récapitulatif d'un noeud affiche également des informations sur le module de noyau DRBD sur lequel repose le gestionnaire de files d'attente de données répliquées. Lorsque vous mettez à niveau un gestionnaire de files d'attente de données répliquées (RDQM), il est important de s'assurer que la version correcte du module de noyau DRBD est installée pour la version du noyau RHEL qui s'exécute sur le système. Le statut affiche la version du noyau du système d'exploitation, la version du noyau pour laquelle le module DRBD a été généré, la version DRBD et le statut de chargement du module de noyau DRBD.

Vous devez être un utilisateur du groupe mqm pour exécuter la commande **rdqmstatus**. Vous pouvez exécuter la commande sur l'un des noeuds de la paire de gestionnaires de files d'attente de données répliquées pour la reprise après incident.

Procédure

- Pour afficher le statut récapitulatif de tous les gestionnaires de files d'attente de données répliquées pour la reprise après incident sur un noeud, exécutez la commande suivante sur ce noeud:

```
rdqmstatus
```

Le statut des gestionnaires de files d'attente de données répliquées pour la reprise après incident sur le noeud s'affiche, par exemple:

```
Node:                               mqhavm07.exampleco.com
OS kernel version:                  5.14.0-362.18.1
DRBD OS kernel version:              5.14.0-362.18.1
DRBD version:                        9.2.7
DRBD kernel module status:           Loaded

Queue manager name:                  DRQM8
Queue manager status:                 Ended immediately
DR role:                              Secondary

Queue manager name:                  DRQM7
Queue manager status:                 Running
DR role:                              Primary
```

Le statut du module de noyau DRBD est l'une des valeurs suivantes:

Chargé

Indique que le module DRBD a été chargé.

Partiellement chargé

Peut se produire lorsque le module DRBD a été chargé, mais ne fonctionne pas correctement en raison d'une non-concordance.

Non chargé

Le module DRBD n'est pas chargé. Cela peut être affiché dans une configuration nouvellement installée, lorsqu'aucun gestionnaire de files d'attente de données répliquées n'a encore été créé.

Non installé

Indique que le module DRBD n'est pas installé, ou que IBM MQ n'a pas pu déterminer la version du noyau du système d'exploitation du module DRBD.

Versión déjà installée encore chargée

Ce statut peut se produire si un nouveau module DRBD est installé alors que le module DRBD existant est en cours d'exécution (c'est-à-dire qu'un gestionnaire de files d'attente RDQM est en cours d'exécution). Le module nouvellement installé est signalé dans le statut, mais il ne s'agit pas du module en cours d'exécution.

- Pour afficher le statut d'un gestionnaire de files d'attente de données répliquées particulier, entrez la commande suivante:

Le tableau suivant récapitule les informations renvoyées.

<i>Tableau 35. Attributs d'état</i>		
attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Etat de gestionnaire de files d'attente	état (tel qu'affiché par dspmq)	Toujours affiché
UC	<i>n.nn%</i>	Affiché uniquement lorsque le gestionnaire de files d'attente de données répliquées sur le noeud en cours a le rôle principal
Mémoire	<i>nnnMo</i>	Affiché uniquement lorsque le gestionnaire de files d'attente de données répliquées sur le noeud en cours a le rôle principal
Système de fichiers du gestionnaire de files d'attente	<i>nnnMo</i> utilisés, <i>n.nGo</i> alloués [<i>n%</i>]	Affiché uniquement lorsque le gestionnaire de files d'attente de données répliquées sur le noeud en cours a le rôle principal
Rôle de reprise après incident	Principal Secondaire Inconnu	Toujours affiché
Statut DR	Normal	Fonctionnement normal
	Synchronisation en cours	La synchronisation est en cours
	Partitionné	Le gestionnaire de files d'attente a été démarré sur les deux noeuds alors que le réseau de réplication de reprise après incident n'est pas disponible
	Système distant non disponible	La connexion à l'autre noeud a été perdue
	Incohérent	Une synchronisation était en cours, mais elle a été interrompue
	Rétablissement de l'instantané	L'utilisateur a choisi de revenir à l'instantané qui a été pris lorsque le gestionnaire de files d'attente est passé à l'état Incohérent.
	Système distant non configuré	L'instance principale du gestionnaire de files d'attente de données répliquées a été configurée, mais aucune instance secondaire n'a été configurée

Tableau 35. Attributs d'état (suite)		
attribut Statut	Valeurs possibles	Lorsqu'il est affiché
	Echec de la négociation	L'un des noeuds a été défini sur la réplication synchrone et l'autre sur la réplication asynchrone
Type de reprise après incident	synchrone ou asynchrone	Toujours affiché
Port de reprise après incident	<i>port_number</i> (port TCP/IP utilisé pour répliquer les données de ce gestionnaire de files d'attente)	Toujours affiché
Adresse IP locale de reprise après incident	Adresse IP locale à partir de laquelle ce gestionnaire de files d'attente est répliqué pour la reprise après incident	Toujours affiché
Adresse IP distante de la reprise après incident	Adresse IP distante vers laquelle ce gestionnaire de files d'attente est répliqué pour la reprise après incident	Toujours affiché
Données non synchronisées de la reprise après incident	<i>nKo</i>	S'affiche lorsque le noeud distant est indisponible ou incohérent
Progression de la synchronisation de reprise après incident	<i>n%</i>	S'affiche lorsque la synchronisation est en cours
Temps estimé de la reprise après incident jusqu'à l'achèvement	<i>AAAA-MM-JJ HH:MM:SS</i>	S'affiche lorsque la synchronisation est en cours
Progression de la réversion d'instantané	<i>n%</i>	S'affiche lorsque le statut de la reprise après incident est <i>Reverting to snapshot</i> . Le nombre de statuts est réduit, de sorte que 0% indique que l'exécution est terminée
Dernière reprise après incident synchronisée	<i>AAAA-MM-JJ HH:MM:SS</i>	S'affiche lorsque les données de reprise après incident ne sont pas synchronisées (après la synchronisation initiale). Indique l'heure et la date de la dernière synchronisation des données.

Exemple

Exemple de statut normal sur le noeud principal:

```

Queue manager status:      Running
CPU:                       0.00
Memory:                    123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
DR role:                   Primary
DR status:                 Normal
DR type:                   Synchronous

```

```
DR port: 3000
DR local IP address: 192.168.20.1
DR remote IP address: 192.168.20.2
```

Exemple de statut normal sur un noeud secondaire:

```
Queue manager status: Ended immediately
DR role: Secondary
DR status: Normal
DR port: 3000
DR local IP address: 192.168.20.2
DR remote IP address: 192.168.20.1
```

Exemple de statut sur le noeud principal lorsque la synchronisation est en cours:

```
Queue manager status: Running
CPU: 0.53
Memory: 124MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
DR role: Primary
DR status: Synchronization in progress
DR type: Synchronous
DR port: 3000
DR local IP address: 192.168.20.1
DR remote IP address: 192.168.20.2
DR synchronization progress: 11.0%
DR estimated time to completion: 2017-09-06 14:55:05
```

Exemple de noeud principal, montrant qu'il est partitionné:

```
Queue manager status: Running
CPU: 0.02
Memory: 124MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
DR role: Primary
DR status: Partitioned
DR type: Synchronous
DR port: 3000
DR local IP address: 192.168.20.1
DR remote IP address: 192.168.20.2
```

Exemple de noeud principal montrant qu'il n'est pas synchronisé avec le noeud secondaire:

```
Queue manager status: Running
CPU: 0.00
Memory: 123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
DR role: Primary
DR status: Remote unavailable
DR type: Asynchronous
DR port: 3000
DR local IP address: 192.168.20.1
DR remote IP address: 192.168.20.2
DR out of sync data: 15932KB
DR last in sync: 2020-07-27 16:01:47
```

Exemple de statut récapitulatif montrant une non-concordance entre la version du noyau du système d'exploitation (RHEL 9.3) et le module de noyau DRBD (ciblé pour RHEL 9.2). Même si le statut signale que le module de noyau DRBD est chargé et que le gestionnaire de files d'attente attendu est en cours d'exécution, vous devez mettre à jour le module de noyau DRBD avec la version ciblée pour le noyau de système d'exploitation en cours d'exécution dans cette situation.

```
Node: mqhvm07.exampleco.com
OS kernel version: 5.14.0-362.18.1
DRBD OS kernel version: 5.14.0-284.11.1
DRBD version: 9.2.7+ptf.14
DRBD kernel module status: Loaded

Queue manager name: DRQM8
Queue manager status: Ended immediately
DR role: Secondary

Queue manager name: DRQM7
```

Queue manager status:	Running
DR role:	Primary

Exemple de statut récapitulatif montrant une non-concordance entre la version de noyau du système d'exploitation (RHEL 8.10) et le module de noyau DRBD (destiné à RHEL 8.8). Dans cet exemple, la non-concordance de version est plus grave et le chargement du module de noyau DRBD échoue. QM3 est un gestionnaire de files d'attente de reprise après incident et doit être l'instance principale, mais comme le module de noyau DRBD n'a pas été entièrement chargé, il est signalé comme étant secondaire avec un statut de reprise après incident Unknown. Pour résoudre cet incident, le module de noyau DRBD doit être mis à jour avec la cible de version du noyau du système d'exploitation en cours d'exécution.

Node:	mqhvm57.exampleco.com
OS kernel version:	4.18.0-553
DRBD OS kernel version:	4.18.0-477
DRBD version:	9.2.7+ptf.14
DRBD kernel module status:	Partially loaded
Queue manager name:	QM3
Queue manager status:	Status not available
DR role:	Secondary
DR status:	Unknown

Référence associée

 [rdqmstatus](#)

Fonctionnement dans un environnement de reprise après incident

Il existe un certain nombre de situations dans lesquelles vous pouvez basculer vers le gestionnaire de files d'attente secondaire dans une configuration de reprise après incident.

Reprise après incident

Suite à la perte définitive du gestionnaire de files d'attente principal sur le site principal, vous démarrez le gestionnaire de files d'attente secondaire sur le site de reprise. Les applications se reconnectent au gestionnaire de files d'attente sur le site de reprise et le gestionnaire de files d'attente secondaire traite les messages des applications. Les étapes permettant de revenir à la configuration précédente dépendent de la cause de l'échec, Par exemple, la perte complète du noeud principal par rapport à la perte temporaire.

Pour connaître les étapes à suivre après une perte temporaire du site principal, voir [«Basculement vers un noeud de reprise»](#), à la page 672. Pour prendre connaissance des étapes à effectuer suite à un échec permanent, voir [«Remplacement d'un noeud défaillant dans une configuration de reprise après incident»](#), à la page 674.

Prise en charge du test de la reprise après incident

Vous pouvez tester la configuration de reprise après incident en basculant temporairement sur l'instance secondaire et en vérifiant que les applications peuvent se connecter correctement. Vous suivez la même procédure que lorsque vous basculez à la suite d'une défaillance temporaire du noeud principal, voir [«Basculement vers un noeud de reprise»](#), à la page 672.

Rétablissement de l'instantané

Si vous rencontrez une défaillance dans le noeud principal alors qu'une synchronisation est en cours, vous pouvez revenir à l'image instantanée des données du gestionnaire de files d'attente secondaire juste avant le début de la synchronisation. Le serveur secondaire est ensuite restauré dans un état cohérent et peut être exécuté en tant que serveur principal. Pour revenir à l'image instantanée, vous devez faire du secondaire le principal, comme décrit dans [«Basculement vers un noeud de reprise»](#), à la page 672. Vous devez vérifier que le rétablissement de l'image instantanée est terminé (à l'aide de la commande **rdqmstatus**) avant de démarrer le gestionnaire de files d'attente.

 [Basculement vers un noeud de reprise](#)

Si un incident survient sur votre site principal, vous pouvez basculer sur votre site de reprise.

Pourquoi et quand exécuter cette tâche

Suite à la perte du gestionnaire de files d'attente principal sur le site principal, vous faites du gestionnaire de files d'attente secondaire sur le site de reprise le gestionnaire de files d'attente principal et le démarrez. Les applications se reconnectent au gestionnaire de files d'attente sur le site de reprise et le gestionnaire de files d'attente traite les messages d'application. Vous pouvez également utiliser cette procédure pour tester votre noeud de reprise.

Important : Vous devez vous assurer que l'instance principale d'un gestionnaire de files d'attente ne peut pas s'exécuter ou qu'elle est arrêtée et transformée en instance secondaire avant de promouvoir l'instance secondaire d'origine. Sinon, les données partitionnées peuvent s'accumuler.

Vous devez être connecté en tant que superutilisateur ou en tant qu'utilisateur appartenant au groupe mqm et disposant de la configuration sudo nécessaire.

Procédure

1. Si vous utilisez cette procédure pour tester votre gestionnaire de files d'attente secondaire (c'est-à-dire que l'instance principale est toujours en cours d'exécution), vous devez arrêter l'instance principale et la redésigner comme instance secondaire:

```
endmqm qmname  
rdqmdr -m qmname -s
```

2. Faites du gestionnaire de files d'attente secondaire le gestionnaire de files d'attente principal en entrant la commande suivante sur le noeud de reprise:

```
rdqmdr -m qmname -p
```

3. Lancez le gestionnaire de files d'attente en entrant la commande :

```
strmqm qmname
```

4. Vérifiez que vos applications se reconnectent au gestionnaire de files d'attente sur le gestionnaire de files d'attente de reprise. Si vous avez défini vos canaux avec une liste de noms de connexion alternatifs, en spécifiant vos gestionnaires de files d'attente principaux et secondaires, vos applications se connectent automatiquement au nouveau gestionnaire de files d'attente principal.

Que faire ensuite

Lorsque le noeud défaillant est restauré, à condition que la liaison entre les deux noeuds fonctionne, le gestionnaire de files d'attente ne peut pas démarrer sur ce noeud car il s'exécute sur le noeud de reprise sur lequel vous avez promu l'instance de gestionnaire de files d'attente secondaire. Pour revenir à un fonctionnement normal, vous devez arrêter le gestionnaire de files d'attente sur le noeud de reprise, puis le promouvoir sur le noeud d'origine vers le rôle principal.

Référence associée

[strmqm](#)

[rdqmdr](#)

Test du gestionnaire de files d'attente de données répliquées de reprise

Vous pouvez tester que l'instance de reprise d'un gestionnaire de files d'attente dans une configuration de reprise après incident de gestionnaire de files d'attente de données répliquées fonctionne correctement sans perturber le site principal.

Pourquoi et quand exécuter cette tâche

Vous testez le gestionnaire de files d'attente de reprise en désactivant l'interface entre les noeuds principal et de reprise. Vous faites du gestionnaire de files d'attente secondaire le gestionnaire de files d'attente principal et vous pouvez ensuite tester le gestionnaire de files d'attente autonome. Une fois le test terminé, vous restaurez l'interface et supprimez le gestionnaire de files d'attente de test. Ensuite, vous recréez le gestionnaire de files d'attente en tant que gestionnaire de files d'attente secondaire dans la configuration de reprise après incident.

Procédure

1. Désactivez la connexion réseau entre le noeud principal et le noeud de reprise.
2. Sur le noeud de reprise, faites du gestionnaire de files d'attente le gestionnaire de files d'attente principal:

```
rdqmdr -m QMname -p
```

Où *nom_gest_files_attente* est le nom du gestionnaire de files d'attente.

3. Démarrez le gestionnaire de files d'attente :

```
strmqm QMname
```

4. Connectez des applications au gestionnaire de files d'attente et vérifiez qu'elles fonctionnent comme prévu.
5. Arrêtez le gestionnaire de files d'attente :

```
endmqm QMname
```

6. Supprimez le gestionnaire de files d'attente :

```
dltmqm QMname
```

7. Restaurez la connexion réseau entre le dispositif principal et le dispositif de reprise.
8. Sur le noeud principal, exécutez la commande suivante pour extraire la commande **crtmqm** que vous avez utilisée lors de la première configuration de la reprise après incident.

```
rdqmdr -d -m QMname
```

9. Exécutez la commande **crtmqm** résultante sur le noeud de reprise pour recréer le gestionnaire de files d'attente secondaire. Le gestionnaire de files d'attente principal sur le noeud principal synchronise ses données avec le gestionnaire de files d'attente secondaire pour le mettre à jour.

Linux *Remplacement d'un noeud défaillant dans une configuration de reprise après incident*

Si vous perdez l'un des noeuds dans une configuration de reprise après incident, vous pouvez remplacer le noeud et restaurer la configuration de reprise après incident en suivant cette procédure.

Pourquoi et quand exécuter cette tâche

Si un incident se produit de sorte que le noeud du site principal ne soit plus réparable, vous pouvez remplacer le noeud défaillant pendant que le gestionnaire de files d'attente s'exécute sur le noeud de reprise, puis restaurer la configuration de reprise après incident d'origine. Le noeud de remplacement doit supposer l'identité du noeud défaillant: le nom et l'adresse IP doivent être identiques.

Vous devez être connecté en tant que superutilisateur ou en tant qu'utilisateur appartenant au groupe mqm et disposant de la configuration sudo nécessaire.

Procédure

En cas de perte du gestionnaire de files d'attente sur le site principal, procédez comme suit :

1. Sur le noeud de reprise, exécutez les commandes suivantes pour que le gestionnaire de files d'attente secondaire assume le rôle principal:

```
rdqmdr -m QMname -p
```

Où *nom_gest_files_attente* est le nom du gestionnaire de files d'attente.

2. Extrayez la commande que vous devrez exécuter sur le noeud principal de remplacement pour reconfigurer la reprise après incident:

```
rdqmdr -m QMname -d
```

Copiez la sortie de cette commande.

3. Exécutez la commande suivante pour démarrer le gestionnaire de files d'attente :

```
strmqm QMname
```

4. Vérifiez que vos applications se reconnectent au gestionnaire de files d'attente sur le noeud de reprise. Si vous avez défini vos canaux avec une liste de noms de connexion alternatifs, en spécifiant vos gestionnaires de files d'attente principaux et secondaires, vos applications se connectent automatiquement au nouveau gestionnaire de files d'attente principal.
5. Remplacez le noeud défaillant sur votre site principal et configurez-le pour qu'il ait le même nom et la même adresse IP que ceux que vous avez utilisés pour la reprise après incident sur le noeud d'origine. Configurez ensuite la reprise après incident en exécutant la commande **crtmqm** que vous avez copiée à l'étape 2. Vous disposez maintenant d'une instance secondaire du gestionnaire de files d'attente et l'instance principale synchronise ses données avec l'instance secondaire.
6. Arrêtez l'instance principale en cours.
7. Une fois la synchronisation terminée, faites en sorte que l'instance principale qui s'exécute sur le noeud de reprise soit à nouveau secondaire:

```
rdqmdr -m QMname -s
```

8. Sur le noeud principal de remplacement, faites de l'instance secondaire du gestionnaire de files d'attente l'instance principale:

```
rdqmdr -m QMname -p
```

9. Sur le noeud principal de remplacement, démarrez le gestionnaire de files d'attente:

```
strmqm QMname
```

A présent, vous avez restauré la configuration telle qu'elle existait avant l'incident sur votre site principal.

Référence associée

[strmqm](#)

[rdqmdr](#)

[endmqm](#)

Résolution d'un problème incohérent dans un gestionnaire de files d'attente de données répliquées

Un statut de reprise après incident **inconsistent** peut être signalé si la synchronisation échoue entre les instances principale et secondaire d'un gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Un état incohérent est signalé sur l'instance secondaire d'un gestionnaire de files d'attente car la connexion de réplication à l'instance principale est perdue lors d'une opération de synchronisation. Vous devrez peut-être prendre des mesures pour résoudre cette situation. Prenez en compte la séquence d'événements suivante:

1. Gestionnaire de files d'attente principal DR synchronisé avec le gestionnaire de files d'attente secondaire DR
2. Lien de réplication perdu entre le principal et le secondaire
3. Liaison de réplication restaurée entre le principal et le secondaire

4. Une resynchronisation se produit lorsque le gestionnaire de files d'attente secondaire de reprise après incident rattrape le gestionnaire de files d'attente principal de reprise après incident. Pendant ce temps, le statut de reprise après incident de synchronisation `in progress` est signalé pour les deux gestionnaires de files d'attente.
5. Si la réplication est ensuite à nouveau perdue lors de la resynchronisation, le statut sur le système secondaire de reprise après incident est signalé comme étant `Inconsistent`.

Si le noeud hébergeant le gestionnaire de files d'attente principal est toujours opérationnel et que le lien de réplication peut être restauré, la resynchronisation se produit automatiquement. L'état incohérent est résolu sans que vous n'ayez à intervenir.

Si le noeud hébergeant le gestionnaire de files d'attente principal n'est plus opérationnel, vous pouvez résoudre l'état incohérent en implémentant une image instantanée sur le gestionnaire de files d'attente secondaire. Cette opération rétablit les données au dernier état correct connu.

Procédure

Pour résoudre un état incohérent:

1. Sur le noeud de reprise, faites de l'instance secondaire l'instance principale:

```
rdqmdr -m qmname -p
```

L'opération de rétablissement de l'image instantanée démarre.

2. Sur le noeud de reprise, vérifiez le statut du gestionnaire de files d'attente pour voir quand l'opération de rétablissement de l'image instantanée est terminée:

```
rdqmstatus -m qmname
```

3. Lorsque le statut du gestionnaire de files d'attente est `Normal`, démarrez le gestionnaire de files d'attente:

```
strmqm qmname
```

Résolution d'un problème de partitionnement (déconnexion cérébrale) dans un gestionnaire de files d'attente de données répliquées pour la reprise après

Un problème partitionné peut se produire si les deux gestionnaires de files d'attente d'une paire de reprise après incident s'exécutent simultanément dans le rôle principal.

Pourquoi et quand exécuter cette tâche

Si vous avez promu l'instance secondaire d'un gestionnaire de files d'attente sur le noeud de reprise alors que l'instance principale d'origine continuait de s'exécuter sur le noeud principal, deux versions du même gestionnaire de files d'attente sont en cours d'exécution, chacune avec sa propre vue des données du gestionnaire de files d'attente. Le statut de reprise après incident du gestionnaire de files d'attente sur chaque noeud est signalé comme `Partitioned`.

Vous devez décider lequel des deux gestionnaires de files d'attente possède la vue la plus correcte des données et conserver cet ensemble tout en supprimant l'autre. Vous utilisez la commande **rdqmdr** pour effectuer cette opération.

Il existe deux procédures. La première décrit la conservation des données du noeud principal, la seconde décrit la conservation des données du noeud de récupération.

Procédure

- Pour conserver les données du gestionnaire de files d'attente sur le noeud principal:
 - a) Assurez-vous que les deux instances de gestionnaire de files d'attente sont arrêtées.
 - b) Indiquez que le gestionnaire de files d'attente sur le noeud de reprise est le gestionnaire secondaire:

```
rdqmdr -m qmname -s
```

c) Indiquez que le gestionnaire de files d'attente sur le noeud principal est le noeud principal:

```
rdqmdr -m qmname -p
```

La synchronisation commence avec les données du gestionnaire de files d'attente sur le noeud principal en cours de copie sur le noeud de reprise.

d) Vérifiez le statut de la synchronisation:

```
rdqmstatus -m qmname
```

e) Une fois la synchronisation terminée, démarrez le gestionnaire de files d'attente sur le noeud principal:

```
strmqm qmname
```

• Pour conserver les données du gestionnaire de files d'attente sur le noeud de reprise:

a) Assurez-vous que les deux instances de gestionnaire de files d'attente sont arrêtées.

b) Indiquez que le gestionnaire de files d'attente sur le noeud principal est le noeud secondaire:

```
rdqmdr -m qmname -s
```

c) Indiquez que le gestionnaire de files d'attente sur le noeud de reprise est le noeud principal:

```
rdqmdr -m qmname -p
```

La synchronisation commence avec les données du gestionnaire de files d'attente sur le noeud de reprise en cours de copie sur le noeud principal.

d) Vérifiez le statut de la synchronisation:

```
rdqmstatus -m qmname
```

e) Une fois la synchronisation terminée, rétrogradez le gestionnaire de files d'attente sur le noeud de reprise:

```
rdqmdr -m qmname -s
```

f) Promouvez le gestionnaire de files d'attente sur le noeud principal et démarrez-le:

```
rdqmdr -m qmname -p  
strmqm qmname
```

Changement des adresses IP dans les configurations de reprise après incident

Si vous modifiez les adresses IP de l'une des interfaces dans une configuration de reprise après incident, la réplication n'est plus possible entre les deux noeuds.

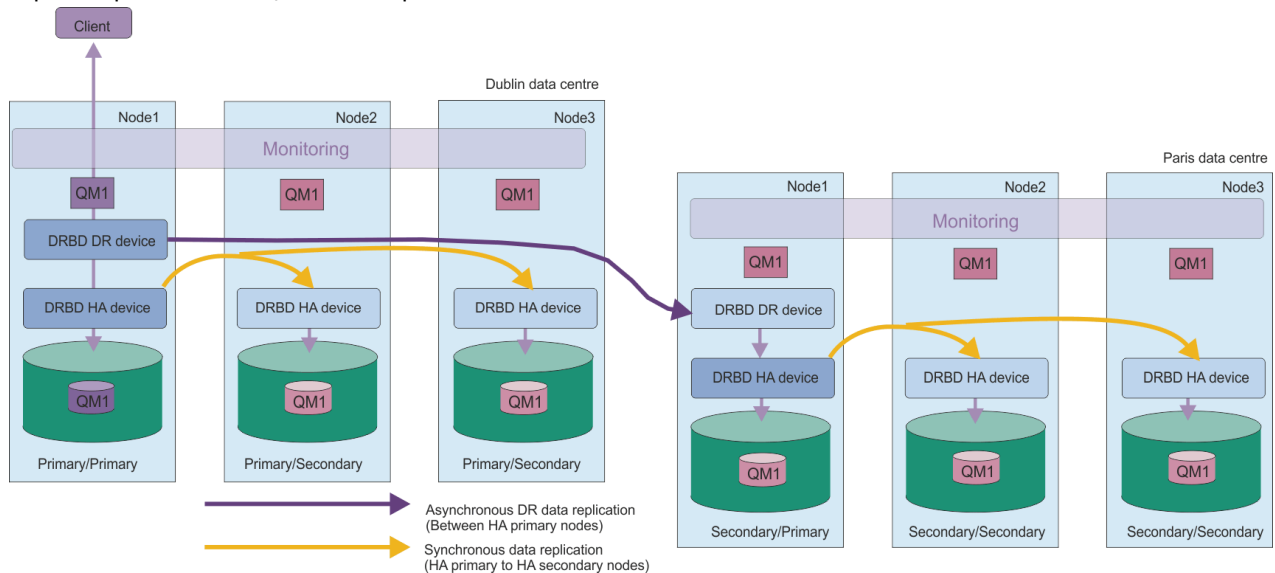
Si vous devez modifier les adresses IP de l'interface de réplication pour l'un de vos noeuds de reprise après incident, vous devez utiliser la procédure suivante:

1. Sur le noeud principal, sauvegardez les gestionnaires de files d'attente de reprise après incident, puis supprimez-les. Sur le noeud de reprise, supprimez les gestionnaires de files d'attente. Voir [«Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ», à la page 737](#) et [«Suppression d'un RDQM de reprise après incident», à la page 658](#).
2. Recréez les gestionnaires de files d'attente de reprise après incident en spécifiant les nouvelles adresses IP et restaurez les sauvegardes (voir [«Création d'un RDQM de reprise après incident», à la page 655](#) et [«Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ», à la page 737](#)).

Vous pouvez configurer un gestionnaire de files d'attente de données répliquées (RDQM) qui s'exécute sur un groupe de haute disponibilité sur un site, mais qui peut basculer sur un autre groupe de haute disponibilité sur un autre site en cas de sinistre qui rend le premier groupe indisponible. On appelle ce type de gestionnaire de files d'attente un gestionnaire de files d'attente de données répliquées de reprise après incident/à haute disponibilité (DR/HA).

Un gestionnaire de files d'attente de données répliquées à haute disponibilité combine les fonctions d'un gestionnaire de files d'attente de données répliquées à haute disponibilité (voir «[Haute disponibilité de RDQM](#)», à la page 616) et d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident (voir «[Reprise après incident RDQM](#)», à la page 650).

Le diagramme suivant illustre un exemple de gestionnaire de files d'attente de données répliquées pour la reprise après incident / haute disponibilité.



La réplication entre les gestionnaires de files d'attente de données répliquées pour la reprise après incident (DR/HA) sur le site principal et le site de reprise après incident est toujours asynchrone. Avec la réplication asynchrone, les opérations telles que IBM MQ PUT ou GET sont terminées et reviennent à l'application avant que l'événement ne soit répliqué sur le gestionnaire de files d'attente secondaire.

Vous pouvez avoir deux sites actifs plutôt que des sites "principaux" et "de reprise", si nécessaire, de sorte que certains de vos gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité s'exécutent sur un site et d'autres sur l'autre lors d'un fonctionnement normal. Si un incident se produit et qu'un site devient indisponible, tous les gestionnaires de files d'attente de données répliquées pour la reprise après incident s'exécutent sur le même groupe à haute disponibilité sur le même site.

Chaque groupe à haute disponibilité est configuré de la même manière qu'un groupe à haute disponibilité ordinaire. Vous pouvez définir des adresses IP flottantes pour un gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité dans chaque groupe à haute disponibilité. L'adresse IP flottante peut être identique ou différente pour chaque groupe HA.

Vous ne pouvez pas mettre à niveau un gestionnaire de files d'attente de données répliquées (RDQM) existant pour qu'il soit un gestionnaire de files d'attente de données répliquées (DR/HA). Vous devez créer un gestionnaire de files d'attente de données (Si nécessaire, vous pouvez sauvegarder les données d'un gestionnaire de files d'attente de données répliquées existant, le supprimer, le recréer en tant que gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité, puis restaurer les données, voir «[Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ](#)», à la page 737.)

Pour configurer des gestionnaires de files d'attente de données répliquées pour la reprise après incident (DR/HA), vous devez effectuer les étapes principales suivantes:

1. Configurez un groupe HA sur le site'main'.
2. Configurez un groupe à haute disponibilité sur le site de reprise.
3. Créez un gestionnaire de files d'attente de données répliquées DR/HA principal / principal sur un noeud du groupe HA sur le site'main'.
4. Créez des gestionnaires de files d'attente de données répliquées pour la reprise après incident (DR/HA) primaires / secondaires sur les deux autres noeuds du site'principal'.
5. Définissez une adresse IP flottante pour une application afin d'accéder au gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA RDQM) lorsqu'elle est exécutée sur l'un des noeuds du groupe à haute disponibilité sur le site principal.
6. Créez un gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA) secondaire / principal sur un noeud du groupe HA sur le site de reprise.
7. Créez des gestionnaires de files d'attente de données répliquées secondaires / DR/HA sur les deux autres noeuds du site de reprise.
8. Définissez une adresse IP flottante pour une application afin d'accéder au gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA RDQM) lorsqu'elle s'exécute sur l'un des noeuds du groupe à haute disponibilité sur le site de reprise.

Des détails sur chacune de ces étapes sont fournis dans les rubriques suivantes.

Linux *Conditions requises pour une solution DR/HA RDQM*

Les exigences pour la solution de gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité sont les mêmes que pour la solution de gestionnaire de files d'attente de données répliquées pour la reprise après incident et la reprise après incident.

Pour plus de détails sur la configuration requise pour les parties à haute disponibilité de la configuration, voir [«Conditions requises pour la solution RDQM HA»](#), à la page 619.

Pour plus de détails sur la partie DR de la configuration, voir [«Conditions requises pour la solution de reprise après incident RDQM»](#), à la page 653.

Linux *Configuration de groupes HA pour les gestionnaires de files d'attente de données répliquées pour la reprise après incident /*

Vous devez créer un groupe à haute disponibilité sur votre site principal et sur votre site de reprise. Si vous disposez d'un groupe à haute disponibilité existant sur l'un des sites, vous pouvez créer des gestionnaires de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité dans ce groupe à haute disponibilité. (Les gestionnaires de files d'attente de données répliquées existants continueront de fonctionner comme auparavant.)

La procédure est la même que celle décrite pour la haute disponibilité des gestionnaires de files d'attente de données répliquées (voir [«Définition du cluster Pacemaker \(groupe HA\)»](#), à la page 623).

Lorsque vous définissez un groupe de haute disponibilité, vous spécifiez les adresses IP utilisées pour la surveillance et la réplication par chaque noeud dans le fichier `rdqm.ini`. Lors de la création d'un groupe à haute disponibilité pour la prise en charge des gestionnaires de files d'attente de données répliquées pour la reprise après incident, vous pouvez également spécifier les adresses IP utilisées pour la réplication de reprise après incident par le groupe à haute disponibilité que vous définissez, ainsi que les adresses IP utilisées pour la réplication de reprise après incident par les noeuds de l'autre groupe à haute disponibilité de la paire de reprise après incident. (Si vous ne spécifiez pas les adresses IP de réplication de reprise après incident dans le fichier `rdqm.ini`, vous pouvez les spécifier dans la ligne de commande lorsque vous créez un gestionnaire de files d'attente de données répliquées pour la reprise après incident.)

Si vous configurez un groupe à haute disponibilité existant, vous pouvez ajouter des adresses IP de réplication de reprise après incident à votre fichier `rdqm.ini` existant. Vous n'avez pas besoin d'exécuter à nouveau `rdqmadm` après la mise à jour de `rdqm.ini`, mais vous devez mettre à jour `rdqm.ini` avant de créer des gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité.

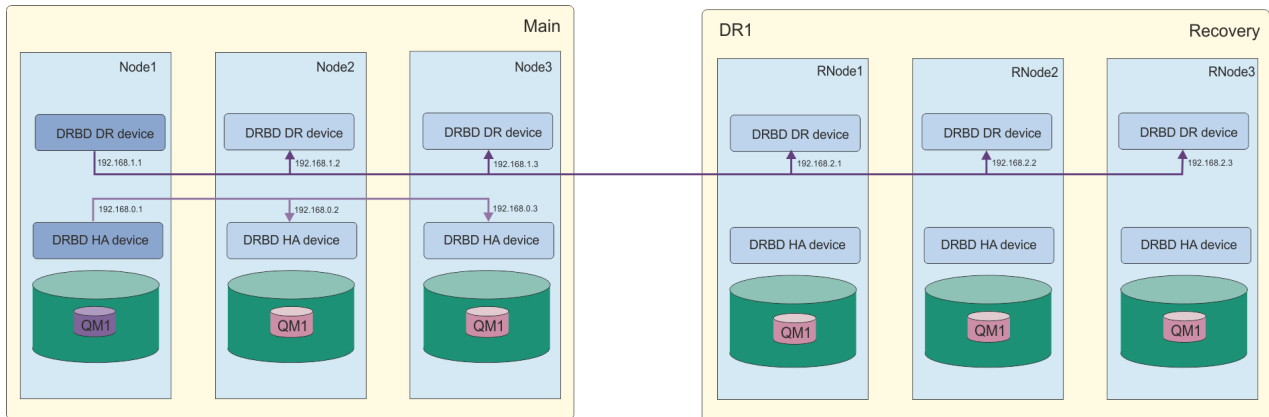
Utilisez l'attribut `DR_Replication` dans les sections `Node` pour spécifier les interfaces de réplication de reprise après incident sur le groupe à haute disponibilité que vous définissez, par exemple:

```
Node:
  Name=Node1
  HA_Replication=192.168.0.1
  DR_Replication=192.168.1.1
Node:
  Name=Node2
  HA_Replication=192.168.0.2
  DR_Replication=192.168.1.2
Node:
  Name=Node3
  HA_Replication=192.168.0.3
  DR_Replication=192.168.1.3
```

Utilisez la section `DRGroup` pour spécifier les adresses de réplication de reprise après incident du groupe HA distant, par exemple:

```
DRGroup:
  Name=DR1
  DR_Replication=192.168.2.1
  DR_Replication=192.168.2.2
  DR_Replication=192.168.2.3
```

Le diagramme suivant illustre cette configuration:



Si vous ne spécifiez pas d'adresses IP de réplication de reprise après incident pour les noeuds du groupe HA local dans le fichier `rdqm.ini` ou sur la ligne de commande lorsque vous créez un gestionnaire de files d'attente de données répliquées pour la reprise après incident, les interfaces `HA_Replication` définies pour chaque noeud sont utilisées pour la réplication de reprise après incident. Vous devez spécifier des adresses de réplication de reprise après incident de groupe à haute disponibilité distante dans le fichier `rdqm.ini` ou sur la ligne de commande `crtmqm`.

Linux *Création de gestionnaires de files d'attente de données répliquées pour la haute disponibilité*

La commande `crtmqm` permet de créer un gestionnaire de files d'attente de données répliquées (RDQM) dans une configuration DR/HA.

Pourquoi et quand exécuter cette tâche

Vous pouvez créer un gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité en tant qu'utilisateur du groupe `mqm` si l'utilisateur peut utiliser `sudo`. Sinon, vous devez créer le gestionnaire de files d'attente de données répliquées en tant que racine.

Vous devez créer un certain nombre de gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité:

- Sur le groupe HA sur le site principal:

- Sur le noeud sur lequel vous souhaitez que le gestionnaire de files d'attente s'exécute dans des conditions normales, créez le gestionnaire de files d'attente de données répliquées principal / principal DR/HA.
- Sur chacun des deux autres noeuds du groupe à haute disponibilité, créez un gestionnaire de files d'attente de données répliquées pour la haute disponibilité (DR/HA) principal / secondaire.
- Sur le groupe à haute disponibilité sur le site de reprise:
 - Sur le noeud sur lequel le gestionnaire de files d'attente s'exécutera en cas de reprise sur le site de reprise, créez le gestionnaire de files d'attente de données répliquées pour la reprise après incident secondaire / principale / à haute disponibilité. Vous pouvez utiliser le résultat de la commande lorsque vous avez créé le gestionnaire de files d'attente principal / principal sur le site principal.
 - Sur chacun des deux autres noeuds du groupe à haute disponibilité, créez un gestionnaire de files d'attente de données répliquées (DR/HA) secondaire / secondaire.

Toutes les instances de gestionnaire de files d'attente doivent avoir le même nom et être affectées à la même quantité de stockage.

Les points suivants fournissent des conseils sur le dimensionnement du système de fichiers du gestionnaire de files d'attente:

1. Lors de la création d'un gestionnaire de files d'attente de données répliquées, un système de fichiers est alloué pour stocker les données et les journaux du gestionnaire de files d'attente. Il est important de dimensionner ce système de fichiers de manière appropriée afin que le gestionnaire de files d'attente puisse enregistrer l'activité en cours dans ses journaux et stocker les messages d'application dans les files d'attente. Lors de la définition de la taille du système de fichiers, tenez compte des pics de besoins en matière de messagerie, de la croissance future de la charge de travail et des indisponibilités d'application susceptibles de générer des messages dans les files d'attente. Pour savoir comment calculer la taille du journal de reprise du gestionnaire de files d'attente, voir [«Quelle est la taille de mon système de fichiers journaux?»](#), à la page 715. Lors du calcul des exigences de stockage pour les messages d'application, la taille et le nombre de messages doivent être pris en compte, ainsi que leur en-tête MQMD et les propriétés de message qu'ils possèdent.
2. Les systèmes de fichiers du gestionnaire de files d'attente de données répliquées ne peuvent pas être redimensionnés dynamiquement. Vous devez sauvegarder puis restaurer un gestionnaire de files d'attente de données répliquées (RDQM) avec un système de fichiers plus grand si nécessaire. Voir [«Redimensionnement du système de fichiers pour un gestionnaire de files d'attente de données répliquées à haute disponibilité»](#), à la page 633.
3. Vous pouvez limiter la taille des files d'attente individuelles sur le disque en utilisant des attributs de file d'attente locale, tels que MAXDEPTH et MAXFSIZE. Voir [Modification des fichiers de file d'attente IBM MQ](#).
4. Vous devez surveiller votre utilisation continue du disque et répondre de manière appropriée si l'utilisation du disque augmente avant que l'utilisation du système de fichiers ne devienne critique. L'utilisation du système de fichiers peut être surveillée à l'aide des fonctions de plateforme / système d'exploitation ou en s'abonnant aux métriques publiées dans les rubriques système IBM MQ décrites dans [Métriques publiées sur les rubriques système](#).

Procédure

- Pour créer le gestionnaire de files d'attente de données répliquées principal / principal DR/HA:
 - a) Entrez la commande suivante :

```

crtmqm -sx -rr p
    [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
    (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -in GroupName)
    -rp DRPort
    [-z] [-q] [-c Text] [-d DefXmitQ] [-h MaxHandles]
    [-g ApplicationGroup] [-oa user|group]
    [-t TrigInt] [-u DeadQ] [-x MaxUMsgs]
    [-lp LogPri] [-ls LogSec]
    [-lc | -ll | -lla | -lln] [-lf LogFileSize]
    [-p Port] [-fs FilesystemSize] QMgrName

```

où :

-sx

Indique que le rôle de haute disponibilité initial est principal.

-rr p

Indique que le rôle de reprise après incident initial est principal.

-rl DRLocalIP1, DRLocalIP2, DRLocalIP3

Vous pouvez éventuellement spécifier les adresses IP des interfaces de reprise après incident sur les trois noeuds du site local (c'est-à-dire le site principal). S'il n'est pas spécifié, les adresses IP spécifiées dans le fichier `rdqm.ini` sont utilisées.

-ri DRRemoteIP1, DRRemoteIP2, DRRemoteIP3

Indiquez les adresses IP des interfaces de reprise après incident sur les trois noeuds du site distant (c'est-à-dire le site de reprise). Vous devez spécifier ce paramètre ou le paramètre `-rn`.

-rn GroupName

Indiquez le nom du groupe HA distant comme indiqué dans le fichier `rdqm.ini`. Vous devez spécifier `-ri` ou `-rn`.

-rp Port

Indique le port à utiliser pour la réplication de reprise après incident.

autre_options_mqm_crtm

Vous pouvez éventuellement spécifier une ou plusieurs des options **crtmqm** générales suivantes:

- -z
- -q
- -c *Texte*
- -d *FileAttenteTransmissionParDéfaut*
- -h *MaxHandles*
- -g *ApplicationGroup*
- -oa utilisateur | groupe
- -t *TrigInt*
- -u *DeadQ*
- -x *MaxUMsgs*
- -lp *LogPri*
- -ls *LogSec*
- -lc | -l
- -lla | -lln
- -lf *LogFileTaille*
- -p *port*

-fs taille

Facultatif. Indique la taille du système de fichiers à créer pour le gestionnaire de files d'attente, c'est-à-dire la taille du volume logique créé dans le groupe de volumes `drbdpool`. Un autre volume logique de cette taille est également créé pour prendre en charge le retour à l'opération d'image instantanée, de sorte que le stockage total pour le gestionnaire de files d'attente de données répliquées pour la reprise après incident est un peu plus de deux fois supérieur à celui spécifié ici.

Taille est une valeur numérique, spécifiée en Go. Vous pouvez indiquer une valeur en Mo en entrant la valeur suivie du caractère M. Par exemple, pour spécifier une taille de système de fichiers de 3 Go, entrez 3. Pour spécifier une taille de système de fichiers de 1024 Mo, entrez 1024M. (Vous pouvez également ajouter un suffixe G pour indiquer explicitement les Go.)

NomGestFile

Indique le nom du gestionnaire de files d'attente de données répliquées. Le nom est dépendant maj/min.

Une fois la commande terminée, elle génère la commande que vous pouvez entrer sur le site de reprise pour créer l'instance secondaire / principale du gestionnaire de files d'attente.

- Pour créer un gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA) primaire / secondaire sur les deux autres noeuds du groupe HA:

a) Entrez la commande suivante sur chaque noeud:

```
citmqm -sxs -rr p
      [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
      (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)
      -rp DRPort
      [-fs FilesystemSize] QMgrName
```

où :

-sxs

Indique que le rôle de haute disponibilité initial est secondaire.

-rr p

Indique que le rôle de reprise après incident initial est principal.

-rl DRLocalIP1, DRLocalIP2, DRLocalIP3

Vous pouvez éventuellement spécifier les adresses IP des interfaces de reprise après incident sur les trois noeuds du site local (c'est-à-dire le site principal). S'il n'est pas spécifié, les adresses IP spécifiées dans le fichier `rdqm.ini` sont utilisées.

-ri DRRemoteIP1, DRRemoteIP2, DRRemoteIP3

Indiquez les adresses IP des interfaces de reprise après incident sur les trois noeuds du site distant (c'est-à-dire le site de reprise). Vous devez spécifier ce paramètre ou le paramètre `-rn`.

-rn GroupName

Indiquez le nom du groupe HA distant comme indiqué dans le fichier `rdqm.ini`. Vous devez spécifier `-ri` ou `-rn`.

-rp Port

Indique le port à utiliser pour la réplification de reprise après incident.

-fs taille

Indique la taille du système de fichiers à créer pour le gestionnaire de files d'attente, c'est-à-dire la taille du volume logique créé dans le groupe de volumes `drbdpool`. Si vous avez spécifié une taille autre que la taille par défaut lors de la création du gestionnaire de files d'attente de données répliquées principal / principal, vous devez spécifier la même valeur ici.

Taille est une valeur numérique, spécifiée en Go. Vous pouvez indiquer une valeur en Mo en entrant la valeur suivie du caractère M. Par exemple, pour spécifier une taille de système de fichiers de 3 Go, entrez 3. Pour spécifier une taille de système de fichiers de 1024 Mo, entrez 1024M. (Vous pouvez également ajouter un suffixe G pour indiquer explicitement les Go.)

NomGestFile

Indique le nom du gestionnaire de files d'attente de données répliquées principal / secondaire. Il doit être identique au nom que vous avez spécifié pour l'instance principale / principale du gestionnaire de files d'attente de données répliquées. Notez que le nom est sensible à la casse.

- Pour créer un gestionnaire de files d'attente de données répliquées (DR/HA) secondaire / principal sur le noeud sur lequel le gestionnaire de files d'attente s'exécutera en cas de basculement sur le site de reprise:

a) Utilisez le résultat de la commande lorsque vous avez créé le principal / principal DR/HA sur le site principal, ou entrez la commande suivante:

```
citmqm -sx -rr s
      [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]
      (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)
```

```
-ip DRPort  
[-fs FilesystemSize] QMgrName
```

-sx

Indique que le rôle de haute disponibilité initial est principal.

-rr s

Indique que le rôle de reprise après incident initial est secondaire.

-rl DRLocalIP1, DRLocalIP2, DRLocalIP3

Vous pouvez éventuellement spécifier les adresses IP des interfaces de reprise après incident sur les trois noeuds du site local (c'est-à-dire le site de reprise). S'il n'est pas spécifié, les adresses IP spécifiées dans le fichier `rdqm.ini` sont utilisées.

-ri DRRemoteIP1, DRRemoteIP2, DRRemoteIP3

Indiquez les adresses IP des interfaces de reprise après incident sur les trois noeuds du site distant (c'est-à-dire le site principal). Vous devez spécifier ce paramètre ou le paramètre `-rn`.

-rn GroupName

Indiquez le nom du groupe HA distant comme indiqué dans le fichier `rdqm.ini`. Vous devez spécifier `-ri` ou `-rn`.

-rp Port

Indique le port à utiliser pour la réplication de reprise après incident.

-fs taille

Facultatif. Indique la taille du système de fichiers à créer pour le gestionnaire de files d'attente, c'est-à-dire la taille du volume logique créé dans le groupe de volumes `drbdpool`. Un autre volume logique de cette taille est également créé pour prendre en charge le retour à l'opération d'image instantanée, de sorte que le stockage total pour le gestionnaire de files d'attente de données répliquées pour la reprise après incident est un peu plus de deux fois supérieur à celui spécifié ici.

NomGestFile

Indique le nom du gestionnaire de files d'attente de données répliquées. Le nom est dépendant maj/min.

- Pour créer un gestionnaire de files d'attente de données répliquées pour la reprise après incident secondaire / secondaire sur les deux autres noeuds du site de reprise:

a) Entrez la commande suivante sur chaque noeud:

```
crtmqm -sxs -rr s  
      [-rl DRLocalIP1,DRLocalIP2,DRLocalIP3]  
      (-ri DRRemoteIP1,DRRemoteIP2,DRRemoteIP3 | -rn GroupName)  
      -ip DRPort  
      [-fs FilesystemSize] QMgrName
```

-sxs

Indique que le rôle de haute disponibilité initial est principal.

-rr s

Indique que le rôle de reprise après incident initial est secondaire.

-rl DRLocalIP1, DRLocalIP2, DRLocalIP3

Vous pouvez éventuellement spécifier les adresses IP des interfaces de reprise après incident sur les trois noeuds du site local. S'il n'est pas spécifié, les adresses IP spécifiées dans le fichier `rdqm.ini` sont utilisées.

-ri DRRemoteIP1, DRRemoteIP2, DRRemoteIP3

Indiquez les adresses IP des interfaces de reprise après incident sur les trois noeuds du site distant. Vous devez spécifier ce paramètre ou le paramètre `-rn`.

-rn GroupName

Indiquez le nom du groupe HA distant comme indiqué dans le fichier `rdqm.ini`. Vous devez spécifier `-ri` ou `-rn`.

-rp Port

Indique le port à utiliser pour la réplication de reprise après incident.

-fs taille

Facultatif. Indique la taille du système de fichiers à créer pour le gestionnaire de files d'attente, c'est-à-dire la taille du volume logique créé dans le groupe de volumes drbdpool. Un autre volume logique de cette taille est également créé pour prendre en charge le retour à l'opération d'image instantanée, de sorte que le stockage total pour le gestionnaire de files d'attente de données répliquées pour la reprise après incident est un peu plus de deux fois supérieur à celui spécifié ici.

NomGestFile

Indique le nom du gestionnaire de files d'attente de données répliquées. Le nom est dépendant maj/min.

Remarque : Lorsque vous créez un gestionnaire de files d'attente de données répliquées, le prochain numéro de port disponible au-dessus de 7000 est alloué pour le lien de réplication à haute disponibilité. S'il est découvert que le port choisi est utilisé par une autre application, la commande **crtmqm** échoue avec l'erreur AMQ6543 et ce port est ajouté à une liste d'exclusion. Vous devez supprimer les instances secondaires du gestionnaire de files d'attente, puis exécuter à nouveau la commande **crtmqm**.

Que faire ensuite

Une fois que vous avez créé tous les gestionnaires de files d'attente de données répliquées pour la reprise après incident (DR/HA), vous devez vérifier le statut sur les instances principale / principale et secondaire / principale pour vous assurer que tous les éléments sont corrects. Utilisez la commande **rdqmstatus** sur les noeuds. Les noeuds doivent afficher le statut normal, comme décrit dans [«Affichage du statut des gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité»](#), à la page 688. S'ils n'affichent pas cet état, supprimez l'instance secondaire / principale et recréez-la en prenant soin d'utiliser les arguments corrects.

Tâches associées

[«Création de gestionnaires de files d'attente de données répliquées pour la haute disponibilité»](#), à la page 680

La commande **crtmqm** permet de créer un gestionnaire de files d'attente de données répliquées (RDQM) dans une configuration DR/HA.

Référence associée

[crtmqm](#)

Linux

Suppression d'un gestionnaire de files d'attente de données répliquées à haute disponibilité
La commande **dltmqm** permet de supprimer un gestionnaire de files d'attente de données répliquées DR/HA (RDQM).

Pourquoi et quand exécuter cette tâche

Vous devez exécuter la commande pour supprimer le gestionnaire de files d'attente de données répliquées sur le noeud principal / principal et sur le noeud secondaire / principal. Le gestionnaire de files d'attente de données répliquées doit d'abord être arrêté. Vous pouvez exécuter la commande en tant qu'utilisateur mqm si cet utilisateur dispose des privilèges sudo nécessaires. Sinon, vous devez exécuter la commande en tant que root.

Procédure

- Pour supprimer un gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité, entrez la commande suivante:

```
dltmqm RDQM_name
```

Référence associée

[dltmqm](#)

Linux **Création d'une adresse IP flottante**

Vous pouvez créer des adresses IP flottantes pour chacun de vos groupes à haute disponibilité dans une configuration de gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité.

Une adresse IP flottante permet à un client d'utiliser la même adresse IP pour un gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité, quel que soit le noeud d'un groupe à haute disponibilité sur lequel il s'exécute. Si vos deux groupes à haute disponibilité possèdent des réseaux privés / isolés pour la connectivité des applications, la même adresse IP flottante peut être définie pour les deux groupes. Vous devez toujours définir cette adresse IP flottante deux fois, mais une fois sur chacun de vos groupes HA.

Vous créez et supprimez des adresses IP flottantes à l'aide de la même méthode que pour un gestionnaire de files d'attente de données répliquées à haute disponibilité. Voir [«Création et suppression d'une adresse IP flottante»](#), à la page 637.

Linux **Démarrage, arrêt et affichage de l'état d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité**

Vous utilisez des variantes des commandes de contrôle IBM MQ standard pour démarrer, arrêter et afficher l'état en cours d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité.

Pourquoi et quand exécuter cette tâche

Vous devez exécuter les commandes qui démarrent, arrêtent et affichent l'état en cours d'un gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA) en tant qu'utilisateur appartenant aux groupes `mqm` et `haclient`.

Vous devez exécuter les commandes pour démarrer et arrêter un gestionnaire de files d'attente sur le noeud principal de ce gestionnaire de files d'attente.

Procédure

- Pour démarrer un gestionnaire de files d'attente de données répliquées, entrez la commande suivante sur le noeud principal du gestionnaire de files d'attente de données répliquées:

```
stmqm qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA) que vous souhaitez démarrer.

Le gestionnaire de files d'attente de données répliquées est démarré et Pacemaker commence à gérer le gestionnaire de files d'attente de données répliquées. Vous devez spécifier l'option `-ns` avec `stmqm` si vous souhaitez spécifier d'autres options `stmqm`.

- Pour arrêter un gestionnaire de files d'attente de données répliquées, entrez la commande suivante sur le noeud principal du gestionnaire de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité:

```
endmqm qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées que vous souhaitez arrêter.

Pacemaker cesse de gérer le gestionnaire de files d'attente de données répliquées, puis le gestionnaire de files d'attente de données répliquées est arrêté. Tous les autres paramètres `endmqm` peuvent être utilisés lors de l'arrêt d'un gestionnaire de files d'attente de données répliquées.

- Pour afficher l'état d'un gestionnaire de files d'attente de données répliquées, entrez la commande suivante:

```
dspmqr -m QMname
```

Les informations d'état qui sont générées varient selon que vous exécutez la commande sur le noeud principal ou secondaire du gestionnaire de files d'attente de données répliquées. S'il est exécuté sur le noeud principal, l'un des messages de statut normal renvoyés par **dspmqr** s'affiche. Si vous exécutez la commande sur un noeud secondaire, l'état `Ended immediately` s'affiche. Par exemple, si **dspmqr** est exécuté sur le noeud RDQM7, les informations suivantes peuvent être renvoyées:

```
QMNAME(DRQM8)          STATUS(Ended immediately)
QMNAME(DRQM7)          STATUS(Running)
```

Vous pouvez utiliser des arguments avec **dspmqr** pour déterminer si un gestionnaire de files d'attente de données répliquées est configuré pour la reprise après incident et s'il s'agit actuellement de l'instance principale ou secondaire:

```
dspmqr -m QMname -o (dr | DR)
```

L'une des réponses suivantes s'affiche:

DRROLE()

Indique que le gestionnaire de files d'attente n'est pas configuré pour la reprise après incident.

DRROLE(Primary)

Indique que le gestionnaire de files d'attente est configuré en tant que serveur principal de reprise après incident.

DRROLE(Secondary)

Indique que le gestionnaire de files d'attente est configuré en tant que serveur secondaire de reprise après incident.

La commande **dspmqr -o all** permet d'afficher les informations de reprise après incident et de haute disponibilité pour les gestionnaires de files d'attente de données répliquées pour la reprise après incident et la haute disponibilité. Par exemple, si vous exécutez **dspmqr -o all** sur le noeud sur lequel s'exécute le gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA RDQM), les informations d'état suivantes s'affichent:

```
QMNAME(TESTQM1)          STATUS(Running) HA(Replicated)
DRROLE(Primary)
```

Référence associée

[dspmqr \(affichage des gestionnaires de files d'attente\)](#)

[endmqm \(arrêt d'un gestionnaire de files d'attente\)](#)

[strmqm \(démarrage d'un gestionnaire de files d'attente\)](#)

Actions de ressource ayant échoué dans les configurations DR/HA

Les actions de ressource ayant échoué se produisent lorsque le composant Pacemaker d'une configuration à haute disponibilité de gestionnaire de files d'attente de données répliquées rencontre un problème avec une ressource sur l'un des noeuds d'un groupe à haute disponibilité.

Des actions de ressource ayant échoué peuvent se produire sur l'une ou l'autre des configurations à haute disponibilité dans une configuration DR/HA de gestionnaire de files d'attente de données répliquées. Vous pouvez utiliser la commande **rdqmstatus** pour afficher les actions de ressource ayant échoué et la commande **rdqmclean** pour les effacer (une fois que la cause de l'échec a été résolue). Le processus est le même que pour les configurations à haute disponibilité des gestionnaires de files d'attente de données répliquées sans le composant de reprise après incident. Pour plus d'informations, voir [«Actions de ressource ayant échoué»](#), à la page 640.

Tâches associées

[«Affichage du statut des gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité»](#), à la page 688

Vous pouvez afficher le statut de haute disponibilité et le rôle de reprise après incident des gestionnaires de files d'attente de données répliquées (RDQM) de la fonction de reprise après incident / haute disponibilité.

«Affichage du statut des groupes RDQM et HA», à la page 641

Vous pouvez afficher le statut du groupe à haute disponibilité et des gestionnaires de files d'attente de données répliquées (RDQM) individuels.

Référence associée

[rdqmclean](#)

[rdqmstatus](#)

Linux

Affichage du statut des gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité

Vous pouvez afficher le statut de haute disponibilité et le rôle de reprise après incident des gestionnaires de files d'attente de données répliquées (RDQM) de la fonction de reprise après incident / haute disponibilité.

Pourquoi et quand exécuter cette tâche

La commande **rdqmstatus** permet d'afficher le statut des gestionnaires de files d'attente de données répliquées individuels ou d'obtenir une vue d'ensemble du statut de tous les gestionnaires de files d'attente de données répliquées connus du groupe à haute disponibilité.

Le statut récapitulatif d'un noeud affiche également des informations sur le module de noyau DRBD sur lequel repose le gestionnaire de files d'attente de données répliquées. Lorsque vous mettez à niveau un gestionnaire de files d'attente de données répliquées (RDQM), il est important de s'assurer que la version correcte du module de noyau DRBD est installée pour la version du noyau RHEL qui s'exécute sur le système. Le statut affiche la version du noyau du système d'exploitation, la version du noyau pour laquelle le module DRBD a été généré, la version DRBD et le statut de chargement du module de noyau DRBD.

Remarque : Notez que, dans une configuration HA/DR, la configuration de la reprise après incident utilise toujours la réplication asynchrone, tandis que la configuration de la haute disponibilité utilise toujours la réplication synchrone. Ces valeurs ne sont pas affichées dans la sortie de la commande `rdqmstatus -m qmgr` dans une configuration HA/DR combinée.

Vous devez être un utilisateur des groupes `mqm` et `haclient` pour exécuter la commande **rdqmstatus**. Vous pouvez exécuter la commande sur n'importe quel noeud de l'un des groupes à haute disponibilité.

Procédure

- Pour afficher le statut récapitulatif d'un noeud et des gestionnaires de files d'attente de données répliquées qui font partie de la configuration à haute disponibilité:

```
rdqmstatus
```

L'identité du noeud sur lequel vous avez exécuté la commande et le statut des gestionnaires de files d'attente de données répliquées dans la configuration à haute disponibilité s'affichent, ainsi que leur rôle de reprise après incident en cours, par exemple:

```
Node:                               main-alice
OS kernel version:                  5.14.0-362.18.1
DRBD OS kernel version:              5.14.0-362.18.1
DRBD version:                       9.2.7
DRBD kernel module status:          Loaded

Queue manager name:                 RDQM1
Queue manager status:                Running elsewhere
HA current location:                 main-charlie
HA preferred location:               main-charlie
HA blocked location:                 None

Queue manager name:                 RDQM9
Queue manager status:                Running elsewhere
```



```

HA current location:      main-bob
HA preferred location:   main-bob
HA blocked location:     None
DR role:                  Primary

Queue manager name:      RDQM7
Queue manager status:    Running
HA current location:     This node
HA preferred location:   This node
HA blocked location:     None
DR role:                  Primary

```

Dans cet exemple, RDQM7 et RDQM8 sont des gestionnaires de files d'attente de données répliquées pour la reprise après incident / à haute disponibilité, tandis que RDQM1 est un gestionnaire de files d'attente de données répliquées pour la reprise après incident, qui n'est pas configuré pour pouvoir basculer sur un site de reprise après incident.

Le statut du module de noyau DRBD est l'une des valeurs suivantes:

Chargé

Indique que le module DRBD a été chargé.

Partiellement chargé

Peut se produire lorsque le module DRBD a été chargé, mais ne fonctionne pas correctement en raison d'une non-concordance.

Non chargé

Le module DRBD n'est pas chargé. Cela peut être affiché dans une configuration nouvellement installée, lorsqu'aucun gestionnaire de files d'attente de données répliquées n'a encore été créé.

Non installé

Indique que le module DRBD n'est pas installé, ou que IBM MQ n'a pas pu déterminer la version du noyau du système d'exploitation du module DRBD.

Versión déjà installée encore chargée

Ce statut peut se produire si un nouveau module DRBD est installé alors que le module DRBD existant est en cours d'exécution (c'est-à-dire qu'un gestionnaire de files d'attente RDQM est en cours d'exécution). Le module nouvellement installé est signalé dans le statut, mais il ne s'agit pas du module en cours d'exécution.

- Pour afficher le statut d'un gestionnaire de files d'attente particulier sur tous les noeuds du groupe à haute disponibilité, entrez la commande suivante:

```
rdqmstatus -m qmname
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées dont vous souhaitez afficher le statut. Le statut du gestionnaire de files d'attente de données répliquées sur le noeud en cours s'affiche, suivi d'un récapitulatif du statut des deux autres noeuds du point de vue du noeud en cours.

- Pour afficher le statut d'un gestionnaire de files d'attente particulier sur tous les noeuds du groupe à haute disponibilité, y compris les détails des actions sur les ressources ayant échoué, entrez la commande suivante:

```
rdqmstatus -m qmname -a
```

où *qmname* est le nom du gestionnaire de files d'attente de données répliquées dont vous souhaitez afficher le statut. Le statut du gestionnaire de files d'attente de données répliquées sur le noeud en cours s'affiche, suivi d'un récapitulatif du statut des deux autres noeuds du point de vue du noeud en cours. Elle est suivie de détails sur les actions de ressource ayant échoué associées au gestionnaire de files d'attente de données répliquées (RDQM).

Le tableau suivant récapitule les informations sur le noeud en cours qui peuvent être renvoyées par la commande `rdqmstatus -m qmname` pour un gestionnaire de files d'attente de données répliquées.

Tableau 36. Statut en cours du noeud

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Nom de noeud	<i>nodeName</i>	Toujours affiché
Etat de gestionnaire de files d'attente	état du gestionnaire de files d'attente (un des états valides pour la commande dspmq)	Toujours affiché
UC	<i>n.nn%</i>	Affiché uniquement lorsque le gestionnaire de files d'attente de données répliquées est en cours d'exécution sur ce noeud
Mémoire	<i>nnnMo</i> utilisés	Affiché uniquement lorsque le gestionnaire de files d'attente de données répliquées est en cours d'exécution sur ce noeud
Système de fichiers du gestionnaire de files d'attente	<i>nnnMo</i> utilisés, <i>y.yGo</i> alloués [z%]	Affiché uniquement lorsque le gestionnaire de files d'attente de données répliquées est en cours d'exécution sur ce noeud
Rôle HA	Principal Secondaire Inconnu	Toujours affiché
Statut de la haute disponibilité	Tous les noeuds en veille Ce noeud en veille Noeuds distants en veille Mixte	Tous les noeuds en veille Noeud en cours en veille Les deux noeuds distants en veille Statut différent pour chaque noeud distant
Contrôle de haute disponibilité	Activé Désactivé Inconnu	Toujours affiché. Indique si le gestionnaire de files d'attente de données répliquées est sous le contrôle Pacemaker
Emplacement préféré de la haute disponibilité	Aucun Ce noeud Inconnu <i>nodeName</i>	Toujours affiché

Tableau 36. Statut en cours du noeud (suite)

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Emplacement bloqué de la haute disponibilité	<p>Aucun-Le gestionnaire de files d'attente n'est pas bloqué sur les noeuds</p> <p>Ce noeud-Le gestionnaire de files d'attente ne peut pas s'exécuter sur le noeud en cours en raison d'une ou de plusieurs actions de ressource ayant échoué</p> <p><i>nodename</i> -Le gestionnaire de files d'attente ne peut pas s'exécuter sur <i>nodename</i> en raison d'une ou de plusieurs actions de ressource ayant échoué</p> <p><i>nodename1, nodename2</i> -Le gestionnaire de files d'attente ne peut pas s'exécuter sur <i>nodename1</i> et <i>nodename2</i> en raison d'une ou de plusieurs actions de ressource ayant échoué</p> <p>Tous les noeuds-Le gestionnaire de files d'attente ne peut pas s'exécuter sur tous les noeuds en raison d'une ou de plusieurs actions de ressource ayant échoué</p>	Toujours affiché
Interface IP flottante à haute disponibilité	<i>interface_name</i>	Toujours affiché
Adresse IP flottante haute disponibilité	<i>IPV4_address</i>	Toujours affiché
Rôle de reprise après incident	<p>Principal</p> <p>Secondaire</p> <p>Secondaire en attente</p> <p>Inconnu</p>	Toujours affiché

Tableau 36. Statut en cours du noeud (suite)

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Statut de la reprise après incident	Normal Synchronisation en cours Partitionné Système distant non disponible Incohérent Rétablissement de l'instantané Système distant non configuré Echec de la négociation	Tout va bien. La synchronisation est en cours. L'utilisateur a démarré la file d'attente gestionnaire sur chaque noeud pendant la Le réseau de réplication DR a été unavailable. La connexion à l'autre noeud a été perdu. Une synchronisation était en cours mais a été interrompue. L'utilisateur a choisi de revenir à l'instantané qui a été pris lors de la le gestionnaire de files d'attente a entré le Etat incohérent. Le serveur principal a été configuré mais le secondaire ne l'a pas. La négociation initiale entre les noeuds principal et secondaire a échoué. Cela peut être dû à des types de réplication incompatibles ou si le noeud secondaire est configuré avec une taille de système de fichiers plus petite.
Statut de la reprise après incident (sur le noeud secondaire à haute disponibilité)	Voir <i>HA_Primary_Node</i>	Affiché sur les noeuds secondaires à haute disponibilité car l'état de la reprise après incident n'est connu que sur le noeud principal à haute disponibilité.
Port de reprise après incident	Port TCP/IP utilisé pour répliquer les données de ce gestionnaire de files d'attente.	Toujours affiché.
Adresse IP locale de reprise après incident	Adresse IP locale utilisée par ce gestionnaire de files d'attente pour la réplication de reprise après incident	Toujours affiché.
Liste d'adresses IP distantes de reprise après incident	Adresses IP distantes que ce gestionnaire de files d'attente utilisera pour la réplication de reprise après incident. Liste séparée par des virgules de trois adresses IP.	Toujours affiché.
Adresse IP distante en cours de la reprise après incident	Adresse IP distante en cours à laquelle ce gestionnaire de files d'attente est connecté pour la réplication de reprise après incident.	Pour un serveur principal à haute disponibilité avec une connexion de reprise après incident active.

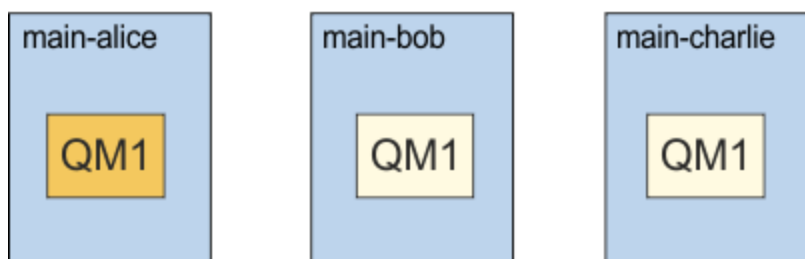
Tableau 36. Statut en cours du noeud (suite)

attribut Statut	Valeurs possibles	Lorsqu'il est affiché
Adresse IP distante en cours de reprise après incident (sur le noeud secondaire à haute disponibilité)	Voir <i>HA_Primary_Node</i>	Affiché sur un noeud secondaire à haute disponibilité car la connexion de reprise après incident se trouve uniquement sur le noeud principal à haute disponibilité
Données non synchronisées de la reprise après incident	xKo	S'affiche lorsque le noeud distant est indisponible ou incohérent.
Progression de la synchronisation de reprise après incident	o%	S'affiche lorsqu'une synchronisation est en cours.
Temps estimé de la reprise après incident jusqu'à l'achèvement	yyyy-MM-dd HH:mm:ss	S'affiche lorsqu'une synchronisation est en cours.
Progression de la réversion d'instantané	o%	S'affiche lorsque le statut de la reprise après incident est "Retour à l'instantané"
Dernière reprise après incident synchronisée	yyyy-MM-dd HH:mm:ss	S'affiche lorsque les données de reprise après incident ne sont pas synchronisées (après la synchronisation initiale). Indique l'heure et la date de la dernière synchronisation des données.

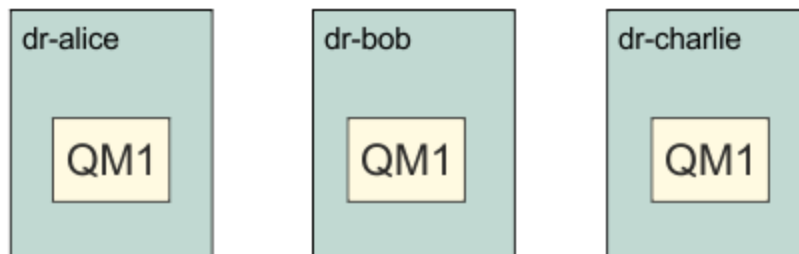
Exemple

Ces exemples illustrent l'exécution de la commande **rdqmstatus -m qm1** sur différents noeuds de la configuration DR/HA suivante:

main site



dr site



Exemple de statut normal sur un noeud qui est le noeud principal de reprise après incident et le noeud principal de haute disponibilité:

```
Node: main-alice
Queue manager status: Running
CPU: 0.00%
Memory: 123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Primary
DR status: Normal
DR port: 3000
DR local IP address: 192.168.1.1
DR remote IP address list: 192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address: 192.168.2.1

Node: main-bob
HA status: Normal

Node: main-charlie
HA status: Normal
```

Exemple de statut normal sur un noeud qui est le noeud principal de reprise après incident et un noeud secondaire de haute disponibilité:

```
Node: main-bob
Queue manager status: Running elsewhere
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: main-alice
HA preferred location: main-alice
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Primary
DR status: See main-alice
DR port: 3000
DR local IP address: 192.168.1.2
DR remote IP address list: 192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address: See main-alice

Node: main-alice
HA status: Normal

Node: main-charlie
HA status: Normal
```

Exemple de statut normal sur un noeud qui est le noeud secondaire de reprise après incident et un noeud principal à haute disponibilité:

```
Node: dr-alice
Queue manager status: Ended immediately
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Secondary
DR status: Normal
DR port: 3000
DR local IP address: 192.168.2.1
DR remote IP address list: 192.168.1.1,192.168.1.2,192.168.1.3
DR current remote IP address: 192.168.1.1

Node: dr-bob
HA status: Normal
```

```
Node: dr-charlie
HA status: Normal
```

Exemple de statut normal sur un noeud qui est le noeud secondaire de reprise après incident et un noeud secondaire de haute disponibilité:

```
Node: dr-bob
Queue manager status: Ended immediately
HA role: Secondary
HA status: Normal
HA control: Enabled
HA current location: dr-alice
HA preferred location: dr-alice
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Secondary
DR status: See dr-alice
DR port: 3000
DR local IP address: 192.168.2.2
DR remote IP address list: 192.168.1.1,192.168.1.2,192.168.1.3
DR current remote IP address: See dr-alice

Node: dr-alice
HA status: Normal

Node: dr-charlie
HA status: Normal
```

Exemple de synchronisation de reprise après incident en cours sur un noeud qui est un noeud principal de reprise après incident et un noeud principal de haute disponibilité:

```
Node: main-alice
Queue manager status: Running
CPU: 0.00%
Memory: 123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Primary
DR status: Normal
DR port: 3000
DR local IP address: 192.168.1.1
DR remote IP address list: 192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address: 192.168.2.1
DR synchronization progress: 11.0%
DR estimated time to completion: 2018-09-06 14:55:05

Node: main-bob
HA status: Normal

Node: main-charlie
HA status: Normal
```

Exemple de reprise après incident partitionnée sur un noeud qui est un noeud principal de reprise après incident et un noeud principal de haute disponibilité:

```
Node: main-alice
Queue manager status: Running
CPU: 0.00%
Memory: 123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
```

```

DR role: Primary
DR status: Partitioned
DR port: 3000
DR local IP address: 192.168.1.1
DR remote IP address list: 192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address: 192.168.2.1
DR out of sync data: 372KB

Node: main-bob
HA status: Normal

Node: main-charlie
HA status: Normal

```

Exemple de reprise après incident désynchronisée sur un noeud qui est un noeud principal de reprise après incident et un noeud principal de haute disponibilité:

```

Node: main-alice
Queue manager status: Running
CPU: 0.00%
Memory: 123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Primary
DR status: Remote unavailable
DR port: 3000
DR local IP address: 192.168.1.1
DR remote IP address list: 192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address: Unknown
DR out of sync data: 372KB
DR last in sync: 2020-02-02 20:22:02

Node: main-bob
HA status: Normal

Node: main-charlie
HA status: Normal

```

Exemple de haute disponibilité désynchronisée sur un noeud qui est un noeud principal de reprise après incident et un noeud principal de haute disponibilité:

```

Node: main-alice
Queue manager status: Running
CPU: 0.00%
Memory: 123MB
Queue manager file system: 51MB used, 1.0GB allocated [5%]
HA role: Primary
HA status: Normal
HA control: Enabled
HA current location: This node
HA preferred location: This node
HA blocked location: None
HA floating IP interface: None
HA floating IP address: None
DR role: Primary
DR status: Normal
DR port: 3000
DR local IP address: 192.168.1.1
DR remote IP address list: 192.168.2.1,192.168.2.2,192.168.2.3
DR current remote IP address: 192.168.2.1

Node: main-bob
HA status: Inconsistent
HA out of sync data: 15932KB
HA last in sync: 2020-02-02 20:22:02

Node: main-charlie
HA status: Normal

```


Exemple de statut récapitulatif montrant une non-concordance entre la version du noyau du système d'exploitation (RHEL 9.3) et le module de noyau DRBD (ciblé pour RHEL 9.2). Même si les rapports de statut indiquent que le module de noyau DRBD est chargé et que le gestionnaire de files d'attente est en cours d'exécution, vous devez mettre à jour le module de noyau DRBD avec la version ciblée pour le noyau de système d'exploitation en cours d'exécution dans cette situation.

```
Node: main-alice
OS kernel version: 5.14.0-362.18.1
DRBD OS kernel version: 5.14.0-284.11.1
DRBD version: 9.2.7+ptf.14
DRBD kernel module status: Loaded

Queue manager name: QM1
Queue manager status: Running
HA current location: This node
HA preferred location: This node
HA blocked location: None
DR role: Primary
```

Exemple de statut récapitulatif montrant une non-concordance entre la version du noyau du système d'exploitation (RHEL 9.3) et le module du noyau DRBD (ciblé pour RHEL 9.0). Dans cet exemple, la non-concordance de version est plus grave et le chargement du module de noyau DRBD échoue. QM1 est un gestionnaire de files d'attente HA/DR qui se déplace vers un autre noeud, son statut de haute disponibilité est inconnu et son statut de reprise après incident est inconnu. Pour résoudre cet incident, le module de noyau DRBD doit être mis à jour avec la cible de version du noyau du système d'exploitation en cours d'exécution.

```
Node: main-alice
OS kernel version: 5.14.0-362.18.1
DRBD OS kernel version: 5.14.0-70.13.1
DRBD version: 9.2.7+ptf.14
DRBD kernel module status: Partially loaded

Queue manager name: QM1
Queue manager status: Running elsewhere
HA status: Unknown
HA current location: main-bob
HA preferred location: This node
HA blocked location: None
DR role: Primary
DR status: Unknown
```

Référence associée

 [rdqmstatus](#)

Fonctionnement dans un environnement DR/HA

Lorsque vous utilisez un environnement DR/HA, vous devez tenir compte de la haute disponibilité et de la reprise après incident.

Si le noeud sur lequel s'exécute un gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA RDQM) échoue, le gestionnaire de files d'attente de données répliquées bascule automatiquement sur un autre noeud de ce groupe à haute disponibilité. Si l'ensemble du site échoue, vous devez démarrer manuellement le gestionnaire de files d'attente de données répliquées (RDQM) sur le noeud préféré dans le groupe HA sur le site de reprise. Les considérations ici sont les mêmes que pour un gestionnaire de files d'attente de données répliquées pour la reprise après incident ordinaire. Pour plus d'informations, voir [«Fonctionnement dans un environnement de reprise après incident»](#), à la page 672.

Si l'un des noeuds échoue complètement et doit être remplacé, voir [«Remplacement d'un noeud défaillant dans une configuration de reprise après incident»](#), à la page 674 et [«Remplacement d'un noeud défaillant dans une configuration à haute disponibilité»](#), à la page 649 pour plus d'informations.

Remplacement d'un noeud défaillant dans une configuration DR/HA

Si l'un des noeuds de l'un de vos groupes à haute disponibilité échoue, vous pouvez le remplacer.

Pourquoi et quand exécuter cette tâche

La procédure varie selon que le noeud que vous remplacez est un noeud principal ou secondaire dans la configuration de reprise après incident. Dans les deux cas, le nouveau noeud doit avoir une configuration identique à celle du noeud que vous remplacez, c'est-à-dire qu'il doit avoir le même nom d'hôte, les mêmes adresses IP, etc.

Vous pouvez également rencontrer la situation où vous avez complètement perdu le groupe HA sur votre site principal ou de reprise et devez remplacer l'intégralité du groupe HA.

Procédure

- Pour un noeud de remplacement qui est un noeud principal dans la configuration de reprise après incident, procédez comme suit sur le nouveau noeud:
 - a) Créez un fichier `rdqm.ini` qui correspond aux fichiers des autres noeuds, puis exécutez la commande `rdqmadm -c` (voir [«Définition du cluster Pacemaker \(groupe HA\)»](#), à la page 623).
 - b) Exécutez la commande `crtmqm -sxs -rr p qmanager` pour recréer chaque gestionnaire de files d'attente de données répliquées pour la reprise après incident / haute disponibilité (voir [«Création de gestionnaires de files d'attente de données répliquées pour la haute disponibilité»](#), à la page 680).
- Pour un noeud de remplacement qui est un noeud secondaire dans la configuration de reprise après incident, procédez comme suit sur le nouveau noeud:
 - a) Créez un fichier `rdqm.ini` qui correspond aux fichiers des autres noeuds, puis exécutez la commande `rdqmadm -c` (voir [«Définition du cluster Pacemaker \(groupe HA\)»](#), à la page 623).
 - b) Exécutez la commande `crtmqm -sx -rr s qmanager` pour recréer chaque gestionnaire de files d'attente de données répliquées pour la reprise après incident / haute disponibilité (voir [«Création de gestionnaires de files d'attente de données répliquées pour la haute disponibilité»](#), à la page 680).
- Pour remplacer la totalité d'un groupe à haute disponibilité, procédez comme suit:
 - a) Si vous perdez la totalité du groupe HA sur le site principal de la reprise après incident (c'est-à-dire le site principal), vous devez suivre les étapes pour effectuer une reprise en ligne gérée sur le site secondaire de la reprise après incident afin de continuer à exécuter vos gestionnaires de files d'attente de données répliquées pour la reprise après incident (voir [«Fonctionnement dans un environnement de reprise après incident»](#), à la page 672). (Si vous perdez la totalité d'un groupe de haute disponibilité sur le site de reprise, vos gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité continuent de s'exécuter sur le site principal sans votre intervention.)
 - b) Recréez le groupe HA sur vos trois noeuds de remplacement, comme décrit dans [«Configuration de groupes HA pour les gestionnaires de files d'attente de données répliquées pour la reprise après incident /»](#), à la page 679.
 - c) Recréez vos gestionnaires de files d'attente de données répliquées pour la reprise après incident / haute disponibilité sur le nouveau groupe à haute disponibilité, comme décrit dans [«Création de gestionnaires de files d'attente de données répliquées pour la haute disponibilité»](#), à la page 680.
 - d) Si nécessaire, effectuez une reprise en ligne gérée depuis votre site de reprise vers votre site principal.

Linux

Exemple de gestionnaire de files d'attente de données répliquées pour la haute disponibilité

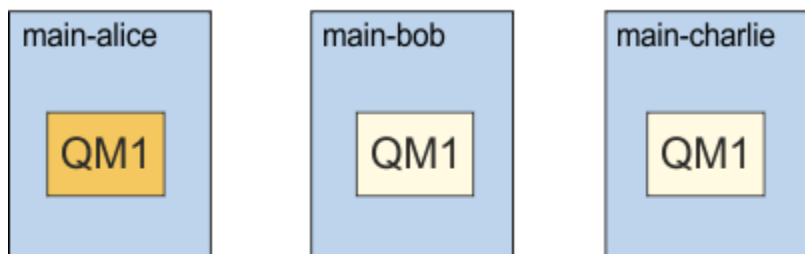
Cet exemple montre comment créer et supprimer un gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA).

Création d'un RDQM DR/HA

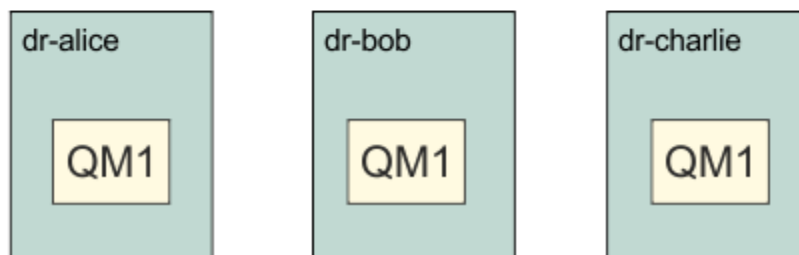
L'exemple de configuration comporte deux sites, nommés 'main' et 'dr'. Chaque site possède trois noeuds, nommés 'alice', 'bob' et 'charlie'. Les noeuds ont un nom complet composé de leur nom de site et de leur nom, donc 'main-alice', 'dr-alice', et ainsi de suite.

Les étapes suivantes créent un gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA) nommé QM1 qui s'exécute sur main-alice. Le noeud main-alice est le noeud principal de haute disponibilité et de reprise après incident.

main site



dr site



Si les adresses IP de reprise après incident locale et distante sont spécifiées dans le fichier `rdqm.ini`, il n'est pas nécessaire de spécifier des adresses IP sur la ligne de commande et un gestionnaire de files d'attente de données répliquées pour la reprise après incident nommé QM1 peut être créé en exécutant la commande suivante sur main-alice:

```
crtmqm -sx -rr p -rn DR1 -rp 7001 QM1
```

Si les adresses IP de reprise après incident locales sont spécifiées dans le fichier `rdqm.ini`, les adresses IP de reprise après incident distantes peuvent être spécifiées sur la ligne de commande:

```
crtmqm -sx -rr p -ri 192.168.2.1,192.168.2.2,192.168.2.3 -rp 7001 QM1
```

Si aucune adresse IP de reprise après incident n'est spécifiée dans le fichier `rdqm.ini`, les adresses IP de reprise après incident distante et locale peuvent être spécifiées sur la ligne de commande:

```
crtmqm -sx -rr p -rl 192.168.1.1,192.168.1.2,192.168.1.3 -ri  
192.168.2.1,192.168.2.2,192.168.2.3 -rp 7001 QM1
```

La sortie en réponse à la création de QM1 est comme illustré dans l'exemple suivant:

```
Creating replicated data queue manager configuration.  
Secondary queue manager created on 'main-bob'.  
Secondary queue manager created on 'main-charlie'.  
IBM MQ queue manager created.  
Directory '/var/mqm/vols/qm1/qmgr/qm1' created.  
The queue manager is associated with installation 'Installation1'.  
Creating or replacing default objects for queue manager 'QM1'.  
Default objects statistics : 83 created. 0 replaced. 0 failed.  
Completing setup.  
Setup completed.
```

```
Enabling replicated data queue manager.  
Replicated data queue manager enabled.  
Issue the following command on the remote HA group to create the DR/HA secondary queue manager:  
crtmqm -sx -rr s -rl 192.168.2.1,192.168.2.2,192.168.2.3 -ri  
192.168.1.1,192.168.1.2,192.168.1.3 -rp 7001 -fs 3072M QM1
```

Copiez la commande à partir du message pour créer l'instance secondaire de reprise après incident de QM1 sur dr-alice:

```
crtmqm -sx -rr s -rl 192.168.2.1,192.168.2.2,192.168.2.3 -ri  
192.168.1.1,192.168.1.2,192.168.1.3 -rp 7001 -fs 3072M QM1
```

Le message suivant est généré sur dr-alice:

```
Creating replicated data queue manager configuration.  
Secondary queue manager created on 'dr-bob'.  
Secondary queue manager created on 'dr-charlie'.  
IBM MQ secondary queue manager created.  
Enabling replicated data queue manager.
```

Tester le serveur secondaire de reprise après incident

Pour tester les fonctions de reprise après incident de QM1, exécutez la commande suivante sur main-alice pour faire de QM1 l'instance secondaire de reprise après incident:

```
rdqmdr -m QM1 -s  
Queue manager 'QM1' has been made the DR secondary on this node.
```

Exécutez la commande suivante sur dr-alice pour faire de QM1 l'instance principale de la reprise après incident sur ce noeud:

```
rdqmdr -m QM1 -p  
Queue manager 'QM1' has been made the DR primary on this node.
```

Suppression d'un gestionnaire de files d'attente de données répliquées à haute disponibilité

Pour supprimer le gestionnaire de files d'attente de données répliquées pour la reprise après incident (DR/HA RDQM) nommé QM1, commencez par arrêter le gestionnaire de files d'attente sur main-alice:

```
endmqm -w QM1  
Replicated data queue manager disabled.  
Waiting for queue manager 'QM1' to end.  
IBM MQ queue manager 'QM1' ended.
```

Exécutez ensuite la commande suivante sur main-alice pour supprimer QM1:

```
dltmqm QM1  
Removing replicated data queue manager configuration.  
Secondary queue manager deleted on 'main-bob'.  
Secondary queue manager deleted on 'main-charlie'.  
IBM MQ queue manager 'QM1' deleted.
```

Enfin, vous devez supprimer QM1 sur dr-alice:

```
dltmqm QM1  
Removing replicated data queue manager configuration.  
Secondary queue manager deleted on 'dr-bob'.  
Secondary queue manager deleted on 'dr-charlie'.  
IBM MQ queue manager 'QM1' deleted.
```

Concepts associés

[«Fonctionnement dans un environnement DR/HA», à la page 697](#)

Lorsque vous utilisez un environnement DR/HA, vous devez tenir compte de la haute disponibilité et de la reprise après incident.

Tâches associées

«Création de gestionnaires de files d'attente de données répliquées pour la haute disponibilité», à la page 680

La commande **crtmqm** permet de créer un gestionnaire de files d'attente de données répliquées (RDQM) dans une configuration DR/HA.

«Suppression d'un gestionnaire de files d'attente de données répliquées à haute disponibilité», à la page 685

La commande **dlmqm** permet de supprimer un gestionnaire de files d'attente de données répliquées DR/HA (RDQM).

CP4I MQ Adv. Native HA

Native HA est une solution à haute disponibilité disponible sur les déploiements de conteneur d' IBM MQ.

Une configuration Native HA se compose de trois noeuds (qui peuvent être, par exemple, trois pods Kubernetes), chacun avec une instance du gestionnaire de files d'attente. Une instance est le gestionnaire de files d'attente actif, qui traite les messages et écrit dans son journal. Chaque fois que le journal est écrit, le gestionnaire de files d'attente actif envoie les données aux deux autres instances, appelées 'répliques'. Chaque réplique écrit dans son propre journal, accuse réception des données, puis met à jour ses propres données de file d'attente à partir du journal répliqué. Si le noeud exécutant le gestionnaire de files d'attente actif échoue, l'une des instances de réplique du gestionnaire de files d'attente prend le rôle actif et dispose des données en cours pour fonctionner.

Pour une présentation détaillée, voir [Native HA](#) dans la section Containers de cette documentation.

La figure ci-après illustre un déploiement type avec trois instances d'un gestionnaire de files d'attente déployé dans trois conteneurs.

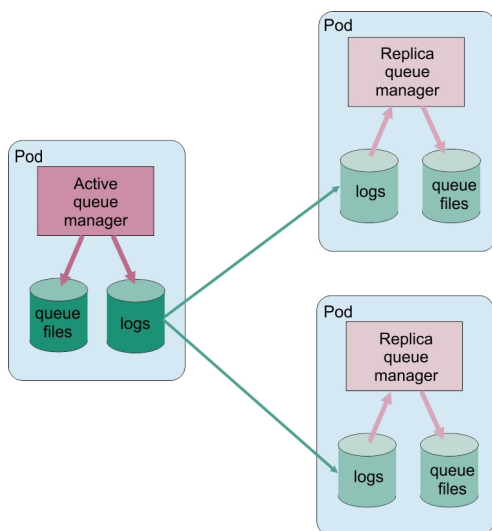


Figure 83. Exemple de configuration Native HA

CP4I MQ Adv. Création de la solution Native HA

La méthode recommandée pour créer une solution Native HA consiste à utiliser IBM MQ Operator. Vous pouvez également créer vos propres conteneurs et configurer manuellement Native HA.

Remarque : Ces informations s'appliquent uniquement aux environnements de conteneur.

Pour créer une solution Native HA à l'aide de IBM MQ Operator, voir [Native HA](#) pour une présentation et [Exemple: configuration d'un gestionnaire de files d'attente Native HA](#) pour des instructions détaillées.

Pour créer vos propres conteneurs et configurer manuellement Native HA, voir [Création du groupe Native HA](#) si vous créez vos propres conteneurs.

Pour IBM MQ dans des conteneurs, vous pouvez utiliser la commande **endmqm** pour arrêter un gestionnaire de files d'attente actif ou de réplique faisant partie d'un groupe Native HA.

Pourquoi et quand exécuter cette tâche

Remarque : Ces informations s'appliquent uniquement aux environnements de conteneur.

La procédure d'arrêt d'un gestionnaire de files d'attente faisant partie d'un groupe Native HA varie selon qu'il s'agit d'une instance active ou d'une instance de réplique. Lorsque vous mettez fin à l'un ou l'autre type d'instance, une vérification est effectuée pour vous assurer que l'arrêt de l'instance ne brise pas le quorum du groupe Native HA. Si le quorum est rompu, la commande **endmqm** échoue.

Lorsque vous émettez une commande **endmqm**, les autres instances du groupe sont averties que cela se produit, de sorte qu'elles ne signalent pas d'erreurs lorsque la connexion est interrompue.

Si une instance active perd le quorum en raison de l'arrêt ou de la déconnexion d'un trop grand nombre d'instances de réplique, l'instance active attend un délai configurable avant de s'arrêter complètement. Cela permet d'arrêter le traitement correctement pendant un certain temps, plutôt que de simplement interrompre les connexions des applications. Cette valeur de délai d'attente peut être spécifiée par l'attribut `QuorumConnectivityTimeout` dans la section `NativeHALocalInstance` du fichier `qm.ini`. La valeur par défaut est 0 secondes.

Procédure

- Pour arrêter l'instance active d'un gestionnaire de files d'attente, exécutez la commande suivante sur le noeud sur lequel l'instance active est en cours d'exécution:

```
endmqm -s QMgrName
```

- Spécifiez l'option `-r` pour aider les applications client à se reconnecter à une autre instance.
- Si cette instance n'est pas l'instance active dans le groupe Native HA, la commande échoue.
- Si l'arrêt de cette instance active entraîne l'échec du quorum de groupe, la commande échoue. (Si d'autres instances se terminent ou deviennent indisponibles en même temps que vous exécutez cette commande, la vérification du quorum risque de ne pas les détecter, le groupe Native HA se termine et ne peut être redémarré que si le nombre d'instances disponibles est suffisant.)

Lorsque le gestionnaire de files d'attente actif s'arrête, l'une des instances de réplique prend le rôle actif. Vous ne pouvez pas spécifier la réplique qui prend le relais, elle est déterminée par la négociation au sein du groupe et dépend de celle qui possède les journaux de transactions les plus à jour.

- Pour arrêter une instance de réplique d'un gestionnaire de files d'attente, exécutez la commande suivante:

```
endmqm -x QMgrName
```

- Si cette instance est l'instance active, la commande échoue.
- Si l'arrêt de cette instance de réplique entraîne l'échec du quorum de groupe, la commande échoue. (Si d'autres instances se terminent ou deviennent indisponibles en même temps que vous exécutez cette commande, la vérification du quorum risque de ne pas les détecter, le groupe Native HA se termine et ne peut être redémarré que si le nombre d'instances disponibles est suffisant.)

Remarque : Vous pouvez également utiliser les commutateurs `-c`, `-i`, `-p` ou `-w` avec la commande **endmqm** sur les instances Native HA, quel que soit leur rôle. L'instance de gestionnaire de files d'attente s'arrête, en ignorant l'effet qui a sur le quorum de groupe. Toutefois, les informations sont toujours partagées avec les autres instances du groupe. Vous pouvez utiliser ces commutateurs avec `-s` pour l'instance active. Vous ne pouvez pas utiliser ces commutateurs avec le commutateur `-x` pour les instances de réplique.

Référence associée

[endmqm \(arrêt d'un gestionnaire de files d'attente\)](#)

Consignation: S'assurer que les messages ne sont pas perdus

IBM MQ enregistre toutes les modifications importantes apportées aux données persistantes contrôlées par le gestionnaire de files d'attente dans un journal de reprise.

Cela inclut la création et la suppression d'objets, les mises à jour de messages persistants, les états de transaction, les modifications apportées aux attributs d'objet et les activités de canal. Le journal contient les informations dont vous avez besoin pour récupérer toutes les mises à jour des files d'attente de messages:

- Conservation des enregistrements des modifications apportées au gestionnaire de files d'attente
- Conservation des enregistrements des mises à jour de file d'attente à utiliser par le processus de redémarrage
- Vous permettre de restaurer des données après une panne matérielle ou logicielle

Toutefois, IBM MQ s'appuie également sur le système de disque qui héberge ses fichiers, y compris les fichiers journaux. Si le système de disques lui-même n'est pas fiable, les informations, y compris les informations de journal, peuvent encore être perdues.



ATTENTION : Vous ne pouvez pas déplacer les journaux de reprise vers un autre système d'exploitation.

A quoi ressemblent les journaux

Les journaux sont constitués de fichiers primaires et secondaires et d'un fichier de contrôle. Vous définissez le nombre et la taille des fichiers journaux et leur emplacement de stockage dans le système de fichiers.

Un journal IBM MQ se compose de deux composants:

1. Un ou plusieurs fichiers de données de journal.
2. Un fichier de contrôle de journal

Un fichier de données de journal est également appelé domaine réservé aux journaux.

Il existe un certain nombre d'extensions de journal qui contiennent les données en cours d'enregistrement. Vous pouvez définir le nombre et la taille (comme expliqué dans [«Section LogDefaults du fichier mq.ini»](#), à la page 106) ou utiliser la valeur par défaut du système, à savoir trois extensions principale et deux extensions secondaires.

Chacune des trois extensions principale et secondaire est définie par défaut sur 16 Mo.

Lorsque vous créez un gestionnaire de files d'attente, le nombre d'extensions de journal préallouées est le nombre d'extensions de journal *principales* allouées. Si vous n'indiquez pas de nombre, la valeur par défaut est utilisée.

IBM MQ utilise deux types de journalisation:

- Circulaire
- Linéaire

Le nombre d'extensions de journal utilisées avec la journalisation linéaire peut être très élevé, en fonction de la fréquence d'enregistrement des images de votre support.

Pour plus d'informations, voir [«Types de journalisation»](#), à la page 704.

ALW

Sur les systèmes IBM MQ for AIX or Linux , si vous n'avez pas modifié le chemin d'accès au journal, les extensions de journal sont créées dans le répertoire:

```
/var/mqm/log/QMgrName
```

Windows

Dans IBM MQ for Windows, si vous n'avez pas modifié le chemin d'accès au journal, les extensions de journal sont créées dans le répertoire:

```
C:\ProgramData\IBM\MQ\log\QMGrName
```

IBM MQ démarre avec ces extensions de journal principal, mais si l'espace de journal principal n'est pas suffisant, il alloue des extensions de journal *secondaires* . Il effectue cette opération de manière dynamique et les supprime lorsque la demande d'espace de journal diminue. Par défaut, jusqu'à deux extensions de journal secondaire peuvent être allouées. Vous pouvez modifier cette allocation par défaut, comme décrit dans «[Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms](#)», à la page 96.

Les extensions de journal sont précédées de la lettre S ou de la lettre R. Les extensions actives, inactives et superflues sont préfixées avec S, tandis que les extensions réutilisées sont préfixées avec R.

Lors de la sauvegarde ou de la restauration de votre gestionnaire de files d'attente, sauvegardez et restaurez toutes les extensions actives, inactives et superflues, ainsi que le fichier de contrôle du journal.

Remarque : Il n'est pas nécessaire de sauvegarder et de restaurer les extensions qui sont réutilisées.

Fichier de contrôle du journal

Le fichier de contrôle de journal contient les informations nécessaires pour décrire l'état des extensions de journal, telles que leur taille et leur emplacement, ainsi que le nom de la prochaine extension disponible.

Important : Le fichier de contrôle des journaux est réservé à l'utilisation du gestionnaire de files d'attente interne.

Le gestionnaire de files d'attente conserve les données de contrôle associées à l'état du journal de reprise dans le fichier de contrôle de journal et vous ne devez pas modifier le contenu du fichier de contrôle de journal.

Le fichier de contrôle du journal se trouve dans le chemin d'accès au journal et est appelé `amqh1c1l.lfh`. Lors de la sauvegarde ou de la restauration de votre gestionnaire de files d'attente, assurez-vous que le fichier de contrôle des journaux est sauvegardé et restauré, ainsi que vos extensions de journal.

Types de journalisation

Dans IBM MQ , il existe deux manières de gérer les enregistrements des activités du gestionnaire de files d'attente: la journalisation avec réutilisation automatique des journaux et la journalisation linéaire. Un troisième type de journalisation, répliqué, est utilisé uniquement par les configurations Native HA.

Journalisation avec réutilisation automatique des journaux

Utilisez la journalisation avec réutilisation automatique des journaux si vous souhaitez uniquement redémarrer la reprise, à l'aide du journal pour annuler les transactions en cours lorsque le système s'est arrêté.

La journalisation circulaire conserve toutes les données de redémarrage dans une boucle de fichiers journaux. La journalisation circulaire remplit le premier fichier de la boucle, puis passe au suivant, et ainsi de suite jusqu'à ce que tous les fichiers soient pleins. Quand le dernier fichier est rempli, le système revient au premier fichier de la boucle et le remplit à nouveau. Ceci continue pendant toute la durée de l'utilisation du programme et vous n'êtes jamais à cours de fichiers journaux.

IBM MQ conserve les entrées de journal requises pour redémarrer le gestionnaire de files d'attente sans perte de données jusqu'à ce qu'elles ne soient plus nécessaires pour assurer la récupération des données du gestionnaire de files d'attente. Le mécanisme de libération des fichiers journaux en vue de leur réutilisation est décrit dans [«Utilisation de l'utilisation de points de contrôle pour garantir une reprise complète»](#), à la page 706.

journalisation linéaire

Utilisez la journalisation linéaire si vous souhaitez à la fois la reprise après incident et la reprise sur support (recréation de données perdues ou endommagées en rélisant le contenu du journal). La journalisation linéaire conserve les données de journal dans une séquence continue de fichiers journaux.

Les fichiers journaux peuvent éventuellement être:

- Réutilisés, mais uniquement lorsqu'ils ne sont plus nécessaires pour la reprise après incident ou la reprise sur incident sur support.
- Archivé manuellement pour le stockage et l'analyse à long terme.

La fréquence des images de support détermine quand les fichiers journaux linéaires peuvent être réutilisés et est un facteur majeur dans la quantité d'espace disque disponible pour les fichiers journaux linéaires.

Vous pouvez configurer le gestionnaire de files d'attente pour qu'il prenne automatiquement des images de support périodiques, en fonction de l'heure ou de l'utilisation du journal, ou vous pouvez planifier des images de support manuellement.

Votre administrateur décide des règles à implémenter et des implications sur l'utilisation de l'espace disque. Les fichiers journaux nécessaires à la reprise par redémarrage doivent toujours être disponibles, tandis que les fichiers journaux nécessaires uniquement à la reprise sur support peuvent être archivés dans un stockage à plus long terme, par exemple sur bande.

Si votre administrateur active la gestion automatique des journaux et les images de support automatiques, la journalisation linéaire se comporte de la même manière qu'un journal circulaire très volumineux, mais avec une redondance améliorée contre les pannes de support activée par la reprise sur support.

Vous pouvez modifier un type de journal existant pour un gestionnaire de files d'attente, de linéaire à circulaire, ou de circulaire à linéaire à l'aide de la commande [migmqlog](#).

Journalisation répliquée

CP4I

Utilisez la journalisation répliquée pour configurer une configuration Native HA. Lorsque vous créez un groupe Native HA, vous créez trois gestionnaires de files d'attente sur des noeuds différents. Vous spécifiez un type de journalisation répliqué avec un nom d'instance unique pour chacun des gestionnaires de files d'attente. La configuration Native HA fournit une solution à haute disponibilité en demandant à une instance active de répliquer les données de journal sur deux instances de réplique. Si l'instance active échoue, l'une des instances de réplique prend le rôle actif. La réplication de journal garantit que peu de données, voire aucune, sont perdues. Voir [«Native HA»](#), à la page 701 pour plus de détails. Un journal répliqué est équivalent à un journal linéaire avec la gestion automatique des journaux et les images de support automatiques activées.

Extensions de journal linéaire qui ne sont pas actives

Multi

Si vous utilisez la gestion automatique des journaux, y compris l'archivage, le consignateur assure le suivi des extensions de journal linéaires qui ne sont pas actives.



Avertissement : Si vous utilisez la gestion automatique des journaux, sans archivage, l'utilisation d'un gestionnaire de files d'attente de sauvegarde n'est pas prise en charge pour ce processus.

Lorsqu'un domaine réservé aux journaux n'est plus nécessaire pour la récupération et, si nécessaire, est archivé, le consignateur, à un point approprié, supprime le domaine réservé aux journaux ou le réutilise.

Une extension de journal réutilisée est renommée pour être la prochaine dans la séquence de journal. Le message AMQ7490 est écrit périodiquement, indiquant combien d'extensions ont été créées, supprimées ou réutilisées.

Le consignateur choisit le nombre d'extensions à conserver prêtes à être réutilisées et le moment où ces extensions doivent être supprimées.

Journal actif

Un certain nombre de fichiers sont dits *actifs* dans la journalisation linéaire et circulaire. Le journal actif correspond à la quantité maximale d'espace de journal, que vous utilisiez la journalisation circulaire ou linéaire, qui peut être référencée par la reprise par redémarrage.

Le nombre de fichiers journaux actifs est généralement inférieur au nombre de fichiers journaux principaux défini dans les fichiers de configuration. (Pour plus d'informations sur la définition du nombre, voir [«Calcul de la taille du journal»](#), à la page 710 .)

Notez que l'espace de journal actif n'inclut pas l'espace requis pour la reprise sur incident lié au support et que le nombre de fichiers journaux utilisés avec la consignment linéaire peut être très élevé, en fonction de votre flux de messages et de la fréquence des images de support.

Journal inactif

Lorsqu'un fichier journal n'est plus nécessaire pour la reprise par redémarrage, il devient *inactif*. Les fichiers journaux qui ne sont pas requis pour la reprise par redémarrage ou la reprise sur support peuvent être considérés comme des fichiers journaux superflus.

Lors de l'utilisation de la gestion automatique des journaux, le gestionnaire de files d'attente contrôle le traitement de ces fichiers journaux superflus. Si vous avez sélectionné la gestion manuelle des journaux, il incombe à votre administrateur de gérer (par exemple, supprimer et archiver) les fichiers journaux superflus s'ils ne présentent plus d'intérêt pour votre opération.

Pour plus d'informations sur la disposition des fichiers journaux, voir [«Gestion des journaux»](#), à la page 717 .

Fichiers journaux secondaires

Bien que les fichiers journaux secondaires soient définis pour la journalisation linéaire, ils ne sont pas utilisés en fonctionnement normal. Si une situation se produit alors que, probablement en raison de transactions de longue durée, il n'est pas possible de libérer un fichier du pool actif car il peut encore être nécessaire pour un redémarrage, les fichiers secondaires sont formatés et ajoutés au pool de fichiers journaux actifs.

Si le nombre de fichiers secondaires disponibles est utilisé, les demandes pour la plupart des autres opérations nécessitant une activité de journal seront refusées avec un code retour MQRC_RESOURCE_PROBLEM renvoyé à l'application et toutes les transactions à exécution longue seront prises en compte pour l'annulation asynchrone.



Avertissement : Tous les types de journalisation peuvent faire face à des pertes d'alimentation inattendues, en supposant qu'il n'y a pas de panne matérielle.

Utilisation de l'utilisation de points de contrôle pour garantir une reprise complète

Les gestionnaires de files d'attente de journalisation avec réutilisation automatique des journaux et de journalisation linéaire prennent en charge la reprise par redémarrage. Quelle que soit la manière dont l'instance précédente du gestionnaire de files d'attente s'arrête brutalement (par exemple, une

panne de courant) au redémarrage, le gestionnaire de files d'attente restaure son état persistant à l'état transactionnel correct au point d'arrêt.

Le redémarrage de la reprise dépend du maintien de l'intégrité du disque. De même, le système d'exploitation doit garantir l'intégrité du disque quel que soit le moment où un arrêt brutal du système d'exploitation peut se produire.

Dans le cas très inhabituel où l'intégrité du disque n'est pas maintenue, la consignation linéaire (et la reprise sur incident) fournit d'autres options de redondance et de capacité de reprise. Avec une technologie de plus en plus courante, telle que RAID, il est de plus en plus rare de subir des problèmes d'intégrité de disque et de nombreuses entreprises configurent la journalisation circulaire et utilisent uniquement la reprise par redémarrage.

IBM MQ est conçu comme un gestionnaire de ressources de journalisation d'écriture anticipée classique. Les mises à jour persistantes des files d'attente de messages sont effectuées en deux étapes:

1. Les enregistrements de journal représentant la mise à jour sont écrits de manière fiable dans le journal de reprise
2. Le fichier de file d'attente ou les mémoires tampon sont mis à jour de la manière la plus efficace pour votre système, mais pas nécessairement de manière cohérente.

Les fichiers journaux peuvent donc devenir plus à jour que la mémoire tampon de la file d'attente sous-jacente et que l'état du fichier.

Si cette situation était autorisée à continuer de manière non anormale, un volume très important de réexécution du journal serait nécessaire pour rendre l'état de la file d'attente cohérent à la suite d'une reprise sur incident.

IBM MQ utilise checkpoints pour limiter le volume de réexécution de journal requis à la suite d'une reprise sur incident. L'événement clé qui contrôle si un fichier journal est actif ou non est un checkpoint.

Un point de contrôle IBM MQ est un point:

- De cohérence entre le journal de reprise et les fichiers d'objet.
- Il s'agit d'un emplacement dans le journal, à partir duquel la réexécution en aval des enregistrements de journal suivants est garantie pour restaurer la file d'attente à l'état logique correct au moment où le gestionnaire de files d'attente peut s'être arrêté.

Lors d'un point de contrôle, IBM MQ vide les anciennes mises à jour des fichiers de files d'attente, selon les besoins, afin de limiter le volume d'enregistrements de journal qui doivent être réexécutés pour ramener les files d'attente à un état cohérent à la suite d'une reprise sur incident.

Le point de contrôle complet le plus récent marque un point dans le journal à partir duquel la réexécution doit être effectuée lors de la reprise sur incident. La fréquence des points de contrôle est donc un compromis entre le temps système d'enregistrement des points de contrôle et l'amélioration du temps de récupération potentiel impliqué par ces points de contrôle.

Le consignateur planifie les points de contrôle plus fréquemment (le suivant est donc planifié avant que le précédent ne soit terminé) car le consignateur tente de conserver le journal actif dans les extensions de journal principal. Si cela n'est pas possible, une erreur [AMQ7466](#) est consignée.

La position dans le journal du début du point de contrôle complet le plus récent est l'un des facteurs clés pour déterminer si un fichier journal est actif ou inactif. L'autre facteur clé est la position dans le journal du premier enregistrement de journal relatif à la première mise à jour persistante effectuée par une transaction active en cours.

Si un nouveau point de contrôle est enregistré dans le deuxième fichier journal ou une version ultérieure et qu'aucune transaction en cours ne fait référence à un enregistrement de journal dans le premier fichier journal, le premier fichier journal devient inactif. Dans le cas de la consignation circulaire, le premier fichier journal est maintenant prêt à être réutilisé. Dans le cas de la consignation linéaire, le premier fichier journal reste généralement requis pour la reprise sur incident lié au support.

Si vous configurez la journalisation avec réutilisation automatique des journaux ou la gestion automatique des journaux, le gestionnaire de files d'attente gère les fichiers journaux inactifs. Si vous configurez la

journalisation linéaire avec la gestion manuelle des journaux, elle devient une tâche d'administration pour gérer les fichiers inactifs en fonction des exigences de votre opération.

IBM MQ génère automatiquement des points de contrôle. Ils sont pris aux moments suivants:

- Lorsque le gestionnaire de files d'attente démarre
- A l'arrêt
- Lorsque l'espace de journalisation est faible
- **Multi** Après la consignation de 50 000 opérations depuis la prise du point de contrôle précédent
- **z/OS** Une fois que *nombre_opérations* a été consigné depuis le point de contrôle précédent, où *nombre_opérations* est le nombre d'opérations défini dans la propriété **LOGLOAD**.

Lorsque IBM MQ redémarre, il trouve l'enregistrement de point de contrôle le plus récent 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. Toutes les opérations qui ont eu lieu depuis le point de contrôle sont relues en avant. C'est ce qu'on appelle la phase de réexécution.

La phase de réexécution ramène les files d'attente à l'état logique dans lequel elles se trouvaient avant l'échec ou l'arrêt du système. Au cours de la phase de réexécution, une liste des transactions qui étaient en cours lors de l'échec ou de l'arrêt du système est créée.

Multi Les messages [AMQ7229](#) et [AMQ7230](#) sont émis pour indiquer la progression de la phase de réexécution.

Afin de savoir quelles opérations doivent être invalidées ou validées, IBM MQ accède à chaque enregistrement de journal actif associé à une transaction en cours. C'est ce qu'on appelle la phase de récupération.

Multi Les messages [AMQ7231](#), [AMQ7232](#) et [AMQ7234](#) sont émis pour indiquer la progression de la phase de récupération.

Une fois que tous les enregistrements de journal nécessaires ont été consultés lors de la phase de reprise, chaque transaction active est à son tour résolue et chaque opération associée à la transaction est annulée ou validée. C'est ce qu'on appelle la phase de résolution.

Multi Le message [AMQ7233](#) est émis pour indiquer la progression de la phase de résolution.

z/OS Sous z/OS, le traitement du redémarrage est constitué de différentes phases.

1. La plage du journal de reprise est établie en fonction de la reprise de support requise pour les ensembles de pages et de l'enregistrement de journal le plus ancien requis pour l'annulation des unités d'oeuvre et l'obtention de verrous pour les unités d'oeuvre en attente de validation.
2. Une fois que la plage de journaux a été déterminée, une lecture de journal en aval est effectuée pour amener les ensembles de pages à l'état le plus récent, ainsi que pour verrouiller les messages liés aux unités de travail en attente de validation ou en cours.
3. Une fois la lecture du journal en aval terminée, les journaux sont lus en amont pour annuler les unités de travail qui étaient en cours ou en cours d'annulation au moment de l'échec.

z/OS Voici un exemple des messages que vous pouvez voir:

```
CSQR001I +MQOX RESTART INITIATED
CSQR003I +MQOX RESTART - PRIOR CHECKPOINT RBA=00000001E48C0A5E
CSQR004I +MQOX RESTART - UR COUNTS - 806
IN COMMIT=0, INDOUBT=0, INFLIGHT=0, IN BACKOUT=0
CSQR030I +MQOX Forward recovery log range 815
from RBA=00000001E45FF7AD to RBA=00000001E48C1882
CSQR005I +MQOX RESTART - FORWARD RECOVERY COMPLETE - 816
IN COMMIT=0, INDOUBT=0
CSQR032I +MQOX Backward recovery log range 817
from RBA=00000001E48C1882 to RBA=00000001E48C1882
CSQR006I +MQOX RESTART - BACKWARD RECOVERY COMPLETE - 818
```

Remarque : S'il y a une grande quantité de journaux à lire, les messages CSQR031I (récupération aval) et CSQR033I (récupération amont) sont émis régulièrement pour indiquer la progression.

Dans [Figure 84](#), à la [page 709](#), tous les enregistrements antérieurs au dernier point de contrôle, Checkpoint 2, ne sont plus requis par IBM MQ. Les files d'attente peuvent être récupérées à partir des informations de point de contrôle et des entrées de journal ultérieures. Pour la journalisation avec réutilisation automatique, tous les fichiers libérés avant le point de contrôle peuvent être réutilisés. Pour un journal linéaire, il n'est plus nécessaire d'accéder aux fichiers journaux libérés pour un fonctionnement normal et de les désactiver. Dans l'exemple, le pointeur de tête de file d'attente est déplacé vers le point de contrôle le plus récent, Checkpoint 2, qui devient alors le nouveau tête de file d'attente, Head 2. Le fichier journal 1 peut désormais être réutilisé.

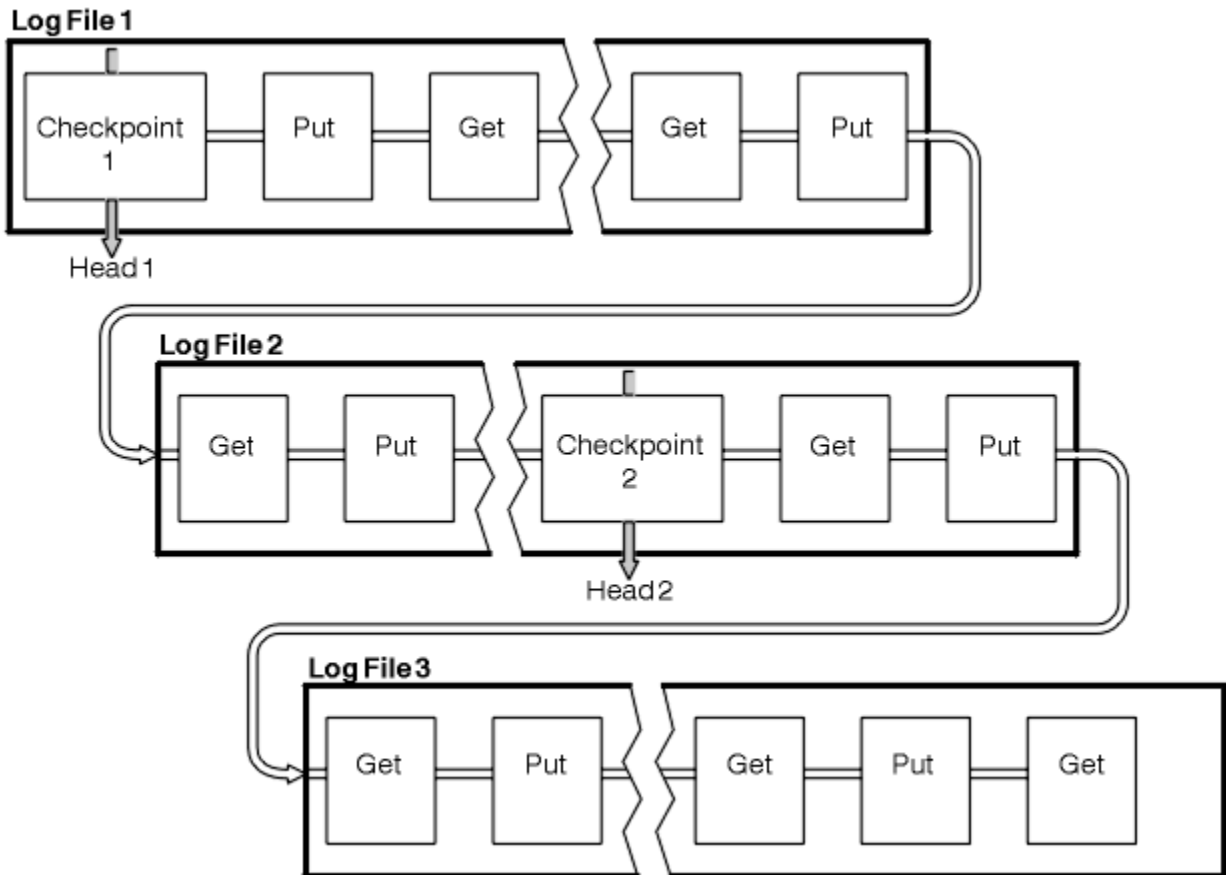


Figure 84. Point de contrôle

Vérification avec points de contrôle avec des transactions à exécution longue

Impact d'une transaction de longue durée sur la réutilisation des fichiers journaux.

[Figure 85](#), à la [page 710](#) montre comment une transaction de longue durée affecte la réutilisation des fichiers journaux. Dans l'exemple, une transaction à exécution longue a créé une entrée dans le journal, affichée sous la forme LR 1, après le premier point de contrôle affiché. La transaction ne se termine (au point LR 2) qu'après le troisième point de contrôle. Toutes les informations de journal à partir de LR 1 sont conservées pour permettre la récupération de cette transaction, si nécessaire, jusqu'à ce qu'elle soit terminée.

Une fois la transaction à exécution longue terminée, au niveau LR 2, la tête du journal passe logiquement au point de contrôle 3, dernier point de contrôle consigné. Les fichiers contenant les enregistrements de journal avant le point de contrôle 3, tête 2, ne sont plus nécessaires. Si vous utilisez la journalisation avec réutilisation automatique, l'espace peut être réutilisé.

Si les fichiers journaux principaux sont complètement saturés avant la fin de la transaction à exécution longue, des fichiers journaux secondaires peuvent être utilisés pour éviter que les journaux ne soient saturés.

Les activités qui sont entièrement sous le contrôle du gestionnaire de files d'attente, par exemple la vérification avec points de contrôle, sont planifiées pour essayer de conserver l'activité dans le journal principal.

Toutefois, lorsque de l'espace de journal secondaire est requis pour prendre en charge un comportement en dehors du contrôle du gestionnaire de files d'attente (par exemple, la durée de l'une de vos transactions), le gestionnaire de files d'attente tente d'utiliser un espace de journal secondaire défini pour permettre à cette activité de se terminer.

Si cette activité ne se termine pas au moment où 80% de l'espace de journal total est utilisé, le gestionnaire de files d'attente lance une action pour récupérer de l'espace de journal, même si cela a un impact sur l'application.

Lorsque la tête de journal est déplacée et que vous utilisez la consignation circulaire, les fichiers journaux principaux peuvent être réutilisables et le consignateur, après avoir rempli le fichier en cours, réutilise le premier fichier principal disponible. Si vous utilisez la journalisation linéaire, la tête de journal est toujours déplacée vers le bas du pool actif et le premier fichier devient inactif. Un nouveau fichier principal est formaté et ajouté au bas du pool en vue de futures activités de journalisation.

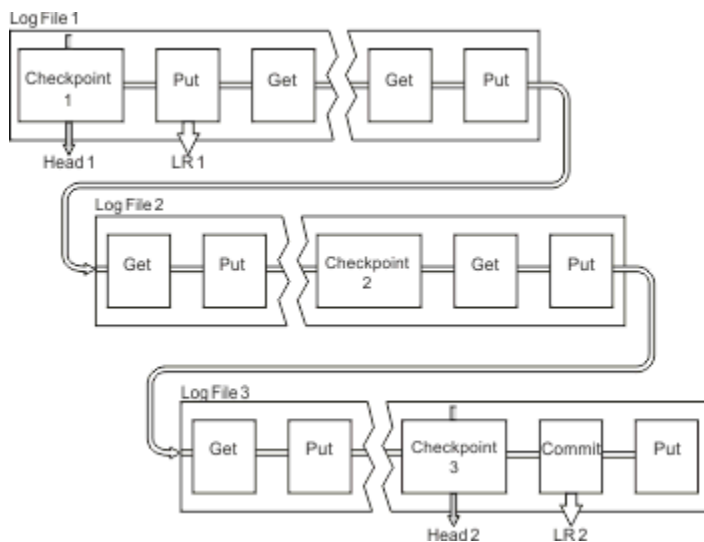


Figure 85. Vérification avec points de contrôle avec une transaction à exécution longue

Calcul de la taille du journal

Estimation de la taille du journal dont un gestionnaire de files d'attente a besoin.

Après avoir déterminé si le gestionnaire de files d'attente utilise la journalisation linéaire ou circulaire, vous devez estimer la taille du journal actif dont le gestionnaire de files d'attente a besoin. La taille du journal actif est déterminée par les paramètres de configuration de journal suivants:

LogFilePages

Taille de chaque fichier journal principal et secondaire en unités de 4K pages

LogPrimaryFiles

Nombre de fichiers journaux primaires préalloués

LogSecondaryFiles

Nombre de fichiers journaux secondaires pouvant être créés pour être utilisés lorsque les fichiers journaux principaux sont saturés

Remarques :

1. Vous pouvez modifier le nombre de fichiers journaux principaux et secondaires chaque fois que le gestionnaire de files d'attente démarre, mais vous risquez de ne pas remarquer immédiatement l'effet de la modification apportée aux journaux secondaires.
2. Vous ne pouvez pas modifier la taille du fichier journal ; vous devez le déterminer **avant** de créer le gestionnaire de files d'attente.
3. Le nombre de fichiers journaux principaux et la taille du fichier journal déterminent la quantité d'espace journal préallouée lors de la création du gestionnaire de files d'attente.
4. Le nombre total de fichiers journaux principaux et secondaires ne peut pas dépasser 511 sur les systèmes AIX and Linux ou 255 sur Windows, ce qui, en présence de transactions de longue durée, limite la quantité maximale d'espace journal disponible pour le gestionnaire de files d'attente pour la reprise après incident. La quantité d'espace journal dont le gestionnaire de files d'attente peut avoir besoin pour la reprise sur incident lié au support ne partage pas cette limite.
5. Lorsque la consignation *circulaire* est utilisée, le gestionnaire de files d'attente réutilise l'espace journal principal et secondaire. Le gestionnaire de files d'attente alloue, jusqu'à une limite, un fichier journal secondaire lorsqu'un fichier journal est saturé et que le fichier journal principal suivant de la séquence n'est pas disponible.

Pour plus d'informations sur le nombre de journaux à allouer, voir [«Quelle est la taille de mon journal actif?»](#), à la page 711 . Les extensions de journal principales sont utilisées dans la séquence et cette séquence ne change pas.

Par exemple, si vous disposez de trois journaux principaux 0, 1 et 2, l'ordre d'utilisation est 0,1,2 suivi de 1,2,0, 2,0,1, retour à 0,1,2 , etc. Tous les journaux secondaires que vous avez alloués sont entrecoupés selon les besoins.

6. Les fichiers journaux principaux sont mis à disposition pour être réutilisés lors d'un point de contrôle. Le gestionnaire de files d'attente prend en compte l'espace de journalisation principal et secondaire avant de prendre un point de contrôle car la quantité d'espace de journalisation est faible.

Le gestionnaire de files d'attente tente de planifier des points de contrôle de manière à conserver l'utilisation des journaux dans les domaines principaux.

Pour plus d'informations, voir [«Section LogDefaults du fichier mq.ini»](#), à la page 106.

Quelle est la taille de mon journal actif?

Estimation de la taille des journaux actifs dont un gestionnaire de files d'attente a besoin.

La taille du journal actif est limitée par:

```
logsize = (primaryfiles + secondaryfiles) * logfilepages * 4096
```

Le journal doit être suffisamment volumineux pour prendre en charge votre plus longue exécution de transaction lorsque le gestionnaire de files d'attente écrit la quantité maximale de données par seconde sur le disque.

Si votre transaction la plus longue s'exécute pendant N secondes et que la quantité maximale de données par seconde écrites sur le disque par le gestionnaire de files d'attente est de B octets par seconde dans le journal, celui-ci doit être au moins:

```
logsize >= 2 * (N+1) * B
```

Le gestionnaire de files d'attente est susceptible d'écrire la quantité maximale de données par seconde sur le disque lorsque vous exécutez la charge de travail maximale ou lorsque vous enregistrez des images de support.

Si une transaction s'exécute si longtemps que le domaine réservé au journal contenant son premier enregistrement de journal n'est pas contenu dans le journal actif, le gestionnaire de files d'attente annule les transactions actives une par une, en commençant par la transaction avec l'enregistrement de journal le plus ancien.

Le gestionnaire de files d'attente doit rendre les anciennes extensions de journal inactives avant que le nombre maximal de fichiers principaux et secondaires ne soit utilisé, et le gestionnaire de files d'attente doit allouer une autre extension de journal.

Déterminez la durée d'exécution de votre transaction la plus longue avant que le gestionnaire de files d'attente ne soit autorisé à l'annuler. Votre transaction en cours d'exécution la plus longue peut être en attente d'un trafic réseau lent ou, dans le cas d'une transaction mal conçue, en attente d'une entrée utilisateur.

Vous pouvez déterminer la durée d'exécution de votre transaction la plus longue en exécutant la commande **runmqsc** suivante:

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

L'émission de la commande `dspmqtin` -a affiche toutes les commandes XA et non XA dans tous les états.

L'exécution de cette commande indique la date et l'heure auxquelles le premier enregistrement de journal a été écrit pour toutes vos transactions en cours.



Avertissement : Pour le calcul de la taille du journal, c'est le temps écoulé depuis l'écriture du premier enregistrement de journal qui est important, et non le temps écoulé depuis le démarrage de l'application ou de la transaction. Arrondissez la longueur de votre transaction la plus longue à la seconde la plus proche. Cela est dû aux optimisations dans le gestionnaire de files d'attente.

Le premier enregistrement de journal peut être écrit longtemps après le démarrage de l'application, si l'application commence par, par exemple, l'émission d'un appel MQGET qui attend un certain temps avant d'obtenir un message.

En examinant la sortie de date et d'heure maximale observée à partir de la

```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

vous avez émis une commande à l'origine, à partir de la date et de l'heure en cours, vous pouvez estimer la durée d'exécution de la transaction la plus longue.

Veillez à exécuter cette commande **runmqsc** à plusieurs reprises pendant que vos transactions les plus longues s'exécutent à une charge de travail maximale afin de ne pas sous-estimer la longueur de votre transaction la plus longue.

Dans IBM MQ 8.0, utilisez les outils du système d'exploitation, par exemple **iostat** sur les plateformes UNIX.

Vous pouvez reconnaître les octets par seconde que le gestionnaire de files d'attente écrit dans le journal en exécutant la commande suivante:

```
amqsrua -m qmgr -c DISK -t Log
```

Octets logiques écrits, affiche les octets par seconde que le gestionnaire de files d'attente écrit dans le journal. Exemple :

```
$ amqsrua -m mark -c DISK -t Log
Publication received PutDate:20160920 PutTime:15383157 Interval:4 minutes,39.579 seconds
Log - bytes in use 37748736
Log - bytes max 50331648
Log file system - bytes in use 316243968
Log file system - bytes max 5368709120
Log - physical bytes written 4334030848 15501948/sec
Log - logical bytes written 3567624710 12760669/sec
Log - write latency 411 uSec
```

Dans cet exemple, le nombre d'octets logiques par seconde écrits dans le journal est 12760669/sec ou environ 12 MiB par seconde.

Utilisation d'


```
DISPLAY CONN(*) UOWLOGDA UOWLOGTI
```

a montré que la transaction la plus longue était:

```
CONN(57E14F6820700069)
EXTCONN(414D51436D61726B20202020202020)
TYPE(CONN)
APPLTAG(msginteg_r)                UOWLOGDA(2016-09-20)
UOWLOGTI(16.44.14)
```

La date et l'heure en cours étant 2016-09-20 16.44.19, cette transaction était en cours d'exécution depuis 5 secondes. Toutefois, vous devez tolérer les transactions qui s'exécutent pendant 10 secondes avant que le gestionnaire de files d'attente ne les annule. Votre taille de journal doit donc être:

```
2 * (10 + 1) * 12 = 264 MiB
```

Le nombre de fichiers journaux doit pouvoir contenir la plus grande taille de journal attendue (calculée dans le texte précédent). Il s'agit des éléments suivants:

Nombre minimal de fichiers journaux = (Taille de journal requise) / (**LogFilePages** * taille de page du fichier journal (4096))

A l'aide de la valeur par défaut **LogFilePages**, qui est 4096, et de l'estimation de la taille du journal de 264MiB, calculée dans le texte précédent, le nombre minimal de fichiers journaux doit être:

```
264MiB / (4096 x 4096) = 16.5
```

c'est-à-dire 17 fichiers journaux.

Si vous dimensionnez votre journal de sorte que la charge de travail attendue s'exécute dans les fichiers principaux:

- Les fichiers secondaires fournissent une certaine éventualité au cas où de l'espace supplémentaire serait nécessaire.
- La journalisation avec réutilisation automatique des journaux utilise toujours des fichiers primaires préalloués, ce qui est légèrement plus rapide que l'allocation et la désallocation de fichiers secondaires.
- Le gestionnaire de files d'attente utilise uniquement l'espace restant dans les fichiers principaux pour calculer quand effectuer le prochain point de contrôle.

Par conséquent, dans l'exemple précédent, définissez les valeurs suivantes pour que la charge de travail s'exécute dans les fichiers journaux principaux:

- **LogFilePages** = 4096
- **LogPrimaryFiles** = 17
- **LogSecondaryFiles** = 5

Notez ce qui suit :

- Dans cet exemple, les données secondaires 5 représentent plus de 20% de l'espace de journal actif.

Le consignateur tente de conserver la charge de travail dans les fichiers principaux uniquement. Par conséquent, le consignateur planifie des points de contrôle lorsqu'une fraction des seuls fichiers principaux est saturée.

Le fait d'avoir les fichiers secondaires est une éventualité, au cas où il y aurait des transactions de longue durée inopinée.

Vous devez savoir que le gestionnaire de files d'attente prend des mesures pour réduire l'utilisation de l'espace journal lorsque plus de 80% de l'espace journal total est utilisé.

- Effectuez le même calcul, que vous utilisiez la journalisation linéaire ou circulaire.

Que vous calculiez la taille d'un journal actif linéaire ou circulaire ne fait aucune différence, car le concept de journal actif signifie le même dans la consignation linéaire et la consignation circulaire.

- Les extensions de journal requises pour la reprise sur support uniquement ne figurent pas dans le journal actif et ne sont donc pas comptées dans le nombre de fichiers principaux et secondaires.
- La zone *LOGUTIL* de DISPLAY QMSTATUS LOG est disponible pour vous aider à calculer, approximativement, la taille du journal actif requise.

Cette zone est conçue pour vous permettre de faire une estimation raisonnable de la taille de journal requise sans échantillonnage permanent afin de déterminer la durée de vos transactions les plus longues ou le débit maximal du gestionnaire de files d'attente.

Quelle est la taille de mes pages LogFile?

En règle générale, faites en sorte que vos pages LogFiles soient suffisamment grandes pour que vous puissiez facilement augmenter la taille de votre journal actif sans atteindre le nombre maximal de fichiers principaux. Un petit nombre de fichiers journaux volumineux est préférable à un grand nombre de petits fichiers journaux car un petit nombre de fichiers journaux volumineux vous permet d'augmenter la taille de votre journal si nécessaire.

Pour la consignation linéaire, des fichiers journaux très volumineux peuvent rendre la variable de performance. Avec des fichiers journaux très volumineux, il est plus important de créer et de formater un nouveau fichier journal ou d'en archiver un ancien. Il s'agit plus d'un problème avec la gestion manuelle et la gestion des journaux d'archivage car avec la gestion automatique des journaux, de nouveaux fichiers journaux sont rarement créés.

Que se passe-t-il si je fais mon journal trop petit?

Points à prendre en compte lors de l'estimation de la taille minimale du journal.

Si vous rendez votre journal trop petit:

- Les transactions à exécution longue seront annulées.
- Le prochain point de contrôle doit commencer avant la fin de la précédente.

Important : Quelle que soit la précision avec laquelle vous estimez la taille de votre journal, l'intégrité des données est maintenue.

Pour plus d'informations sur les points de contrôle, voir «Utilisation de l'utilisation de points de contrôle pour garantir une reprise complète», à la page 706 . Si la quantité d'espace de journal restant dans les extensions de journal actives est insuffisante, le gestionnaire de files d'attente planifie les points de contrôle plus fréquemment.

Un point de contrôle prend un certain temps ; il n'est pas instantané. Plus le nombre de données à enregistrer dans le point de contrôle est élevé, plus le point de contrôle est long. Si le journal est de petits points de contrôle peuvent se chevaucher, ce qui signifie que le point de contrôle suivant est demandé avant la fin du point de contrôle précédent. Si cela se produit, des messages d'erreur sont écrits.

Si des transactions à exécution longue sont annulées ou que des points de contrôle se chevauchent, le gestionnaire de files d'attente poursuit le traitement de la charge de travail. Les transactions de courte durée continuent de s'exécuter normalement.

Toutefois, le gestionnaire de files d'attente ne s'exécute pas de manière optimale et les performances peuvent être dégradées. Vous devez redémarrer le gestionnaire de files d'attente avec un espace de journalisation suffisant.

Que se passe-t-il si je rend mon journal trop grand?

Points à prendre en compte lors de l'estimation de la taille maximale du journal.

Si votre journal est trop volumineux:

- Vous pouvez augmenter le temps nécessaire à un redémarrage d'urgence, bien que cela soit peu probable.
- Vous utilisez de l'espace disque inutile.

- Les transactions à exécution très longue sont tolérées.

Important : Quelle que soit la précision avec laquelle vous estimez la taille de votre journal, l'intégrité des données est maintenue.

Pour vous aider à estimer la taille maximale du journal, vous pouvez utiliser les statistiques d'utilisation du journal. Pour plus d'informations, voir [«Comment définir IMGLOGLN et IMGINTVL»](#), à la page 721 et ALTER QMGR.

Voir [«Utilisation de l'utilisation de points de contrôle pour garantir une reprise complète»](#), à la page 706 pour une description de la façon dont le gestionnaire de files d'attente lit le journal au redémarrage. Le gestionnaire de files d'attente réexécute le journal à partir du dernier point de contrôle, puis résout toutes les transactions qui étaient actives à l'arrêt du gestionnaire de files d'attente.

Pour résoudre une transaction, le gestionnaire de files d'attente lit à nouveau tous les enregistrements de journal associés à cette transaction. Ces enregistrements de journal peuvent être antérieurs au dernier point de contrôle.

En allouant au gestionnaire de files d'attente un journal très volumineux, vous lui donnez le droit de lire chaque enregistrement de journal dans le journal au redémarrage, bien que le gestionnaire de files d'attente n'ait généralement pas à le faire. Potentiellement, dans le cas peu probable où cela se produise, ce processus pourrait prendre beaucoup de temps.

Si la vérification avec points de contrôle s'est arrêtée de manière inattendue avant l'arrêt du gestionnaire de files d'attente, cela augmente considérablement le temps de redémarrage d'un gestionnaire de files d'attente avec un journal volumineux. La limitation de la taille du journal limite le temps de redémarrage d'urgence.

Pour éviter ces problèmes, vous devez vous assurer que:

- Votre charge de travail peut s'insérer confortablement dans un journal qui n'est pas trop volumineux.
- Vous évitez les transactions à exécution longue.

Quelle est la taille de mon système de fichiers journaux?

Estimation de la taille du système de fichiers journaux dont un gestionnaire de files d'attente a besoin.

Il est important que votre système de fichiers journaux soit suffisamment grand pour que votre gestionnaire de files d'attente dispose de suffisamment d'espace pour écrire son journal. Si le gestionnaire de files d'attente remplit complètement le système de fichiers journaux, il écrit des fichiers FFDC, annule des transactions et peut arrêter brutalement le gestionnaire de files d'attente.

La quantité d'espace disque que vous réservez pour votre journal doit être au moins aussi grande que le journal actif. La taille exacte dépend des éléments suivants:

- Votre choix de type de journal (linéaire ou circulaire)
- Taille du journal actif (fichiers principaux, fichiers secondaires, pages de fichier journal)
- Votre choix de gestion des journaux (manuelle, automatique ou d'archivage)
- Vos plans d'urgence dans le cas d'un objet endommagé.

Si vous choisissez un journal circulaire, votre système de fichiers de journal doit être

```
LogFilesystemSize >= (PrimaryFiles + SecondaryFiles + 1) * LogFileSize
```

Cela permet au gestionnaire de files d'attente d'écrire dans tous les fichiers principaux et secondaires. Dans des circonstances exceptionnelles, le gestionnaire de files d'attente peut écrire une extension supplémentaire au-delà du nombre de secondaires. L'algorithme précédent en tient compte.

Si vous choisissez un journal linéaire, le système de fichiers du journal doit être nettement plus grand que le journal actif.

Si vous choisissez la gestion manuelle des journaux, le gestionnaire de files d'attente continue d'écrire dans les nouvelles extensions de journal en fonction de ses besoins et il vous incombe de les supprimer (et de les archiver) lorsqu'elles ne sont plus nécessaires.

La taille du système de fichiers journaux dépend en grande partie de votre stratégie de suppression des extensions superflues ou inactives.

Vous pouvez décider d'archiver et de supprimer des extensions dès qu'elles deviennent inactives (non nécessaires pour la reprise par redémarrage) ou vous pouvez décider d'archiver et de supprimer uniquement les extensions superflues (non nécessaires pour le support ou la reprise par redémarrage).

Si vous archivez et supprimez uniquement les extensions superflues, et si vous avez un objet endommagé, **MEDIALOG** n'ira pas plus loin, de sorte qu'aucune autre extension ne deviendra superflue. Vous allez arrêter d'archiver et de supprimer des extensions jusqu'à ce que vous résolviez le problème, peut-être en récupérant l'objet.

A moins que vous n'arrêtiez la charge de travail, le temps dont vous avez besoin pour résoudre le problème dépend de la taille de votre système de fichiers journaux. Par conséquent, il est recommandé d'avoir un système de fichiers de journal généreux lors de l'utilisation de la journalisation linéaire.

Si vous choisissez un journal linéaire et une gestion automatique ou d'archivage des journaux, le gestionnaire de files d'attente réutilise les extensions de journal.

Les extensions de journal disponibles pour être réutilisées sont précédées de la lettre R. Lorsqu'une image de support est enregistrée, à mesure que des extensions superflues sont archivées, le gestionnaire de files d'attente peut alors réutiliser ces extensions.

Par conséquent, les extensions de réutilisation sont inférieures à la longueur des données écrites dans le journal entre les images de support:

```
ReuseExtents <= LogDataLengthBetweenMediaImages
```

Lors de l'enregistrement automatique des images de support et de la définition de **IMGLOGLN**, `LogDataLengthBetweenMediaImages` peut être jusqu'à deux fois **IMGLOGLN** car **IMGLOGLN** est une cible qui n'est pas un maximum fixe.

Lorsque vous enregistrez manuellement des images de support ou que vous les enregistrez automatiquement par intervalle, `LogDataLengthBetweenMediaImages` dépend de votre charge de travail et de l'intervalle entre les prises d'images.

Outre les extensions actives et les extensions de réutilisation, il existe des extensions inactives (nécessaires pour la reprise sur incident lié au support uniquement) et des extensions superflues (non nécessaires pour le redémarrage ou la reprise sur incident lié au support).

Lors de l'utilisation de la gestion automatique ou de la gestion des journaux d'archivage, le gestionnaire de files d'attente ne réutilise pas les domaines nécessaires à la reprise sur incident lié au support. Par conséquent, le nombre d'extensions inactives dépend de la fréquence à laquelle vous prenez des images de support et de la fréquence à laquelle vous les prenez manuellement ou automatiquement.

IMGINTVL et **IMGLOGLN** sont des cibles, pas un minimum ou un maximum fixe entre les images de support. Toutefois, lors de l'estimation de la taille maximale du système de fichiers journaux dont vous pourriez avoir besoin, il est peu probable que les images de support automatiques soient enregistrées à plus de deux fois **IMGINTVL** ou **IMGLOGLN**.

Lors de la définition de la taille de votre système de fichiers journaux à l'aide de la gestion automatique ou de la gestion des journaux d'archivage, vous devez également tenir compte de ce qui peut se produire si une file d'attente ou un autre objet est endommagé. Dans ce cas, le gestionnaire de files d'attente ne peut pas prendre une image de support de l'objet endommagé et **MEDIALOG** ne peut pas avancer.

Si votre charge de travail continue, votre journal inactif ne sera pas restreint car le domaine le plus ancien nécessaire à la reprise sur incident sur support est encore nécessaire et ne peut pas être réutilisé. Si votre charge de travail continue, vous aurez jusqu'à ce que votre système de fichiers journaux se remplisse complètement pour résoudre le problème, avant que le gestionnaire de files d'attente ne commence à rétrograder les transactions et puisse même s'arrêter brutalement.

Par conséquent, pour la gestion automatique et la gestion des journaux d'archivage:


```
LogFilesystemSize > (PrimaryFiles + SecondaryFiles +
```

```
((TimeBetweenMediaImages *2) + TimeNeededToResolveDamagedObject) * ExtentsUsedPerHour))
* LogFilePages
```

Remarque : L'algorithme précédent suppose que **SET LOG ARCHIVED** est appelé pour chaque domaine, dès qu'il n'est plus nécessaire pour la récupération des supports, pour la gestion des journaux d'archivage.

Gestion des journaux


Le produit prend en charge la gestion automatique des journaux et la reprise automatique des supports des journaux linéaires. Les journaux circulaires sont presque auto-gérés, mais nécessitent parfois une intervention pour résoudre les problèmes d'espace.

Remarque :  La gestion automatique et la gestion des journaux d'archivage ne sont pas valides sous IBM i.

Avec la consignation circulaire, le gestionnaire de files d'attente récupère l'espace disponible des fichiers journaux. Cette activité n'est pas visible par l'utilisateur et vous ne voyez généralement pas la réduction de l'espace disque utilisé car l'espace alloué est rapidement réutilisé.

Vous pouvez supprimer des fichiers secondaires lors de l'utilisation de la journalisation avec réutilisation automatique des journaux. Pour plus d'informations, voir [RESET QMGR TYPE \(REDUCELOG\)](#).

Avec la consignation linéaire, le journal risque d'être saturé si aucun point de contrôle n'a été effectué pendant une longue période ou que la consignation d'un enregistrement de journal par une transaction à exécution longue remonte à longtemps. Le gestionnaire de files d'attente tente d'effectuer des points de contrôle assez souvent pour éviter le premier problème.

 Si le journal est saturé, le message AMQ7463 est généré. En outre, si le journal est saturé car une transaction à exécution longue a empêché la libération d'espace, le message AMQ7465 est généré.

Parmi les enregistrements de journal, seuls ceux consignés depuis le début du dernier point de contrôle complet et ceux consignés par des transactions actives sont requis pour redémarrer le gestionnaire de files d'attente.

Au fil du temps, les enregistrements de journal consignés les plus anciens deviennent inutiles pour redémarrer le gestionnaire de files d'attente.

Si une transaction à exécution longue est détectée, l'activité est planifiée pour annuler cette transaction de manière asynchrone. Si, pour une raison inattendue, cette annulation asynchrone échouait, des appels MQI renverraient MQRC_RESOURCE_PROBLEM.

Notez que l'espace est réservé pour valider ou annuler toutes les transactions en cours. Par conséquent, **MQCMIT** ou **MQBACK** ne doit pas échouer.

Une application dont une transaction est annulée de cette manière ne peut pas effectuer les opérations **MQPUT** ou **MQGET** suivantes en spécifiant un point de synchronisation sous la même transaction.

Toute tentative d'insérer ou d'extraire un message sous un point de synchronisation dans cet état renvoie QRC_BACKED_OUT. L'application peut ensuite émettre **MQCMIT**, qui renvoie MQRC_BACKED_OUT, ou **MQBACK** et démarrer une nouvelle transaction. Si la transaction qui consomme trop d'espace de journal a été annulée, l'espace de journal est libéré et le gestionnaire de files d'attente continue de fonctionner normalement.

Que se passe-t-il lorsqu'un disque est saturé?

Lorsqu'un gestionnaire de files d'attente est configuré pour utiliser la consignation linéaire, le composant de consignation du gestionnaire de files d'attente réagit à une condition de disque saturé de la manière suivante.

Si le disque contenant les fichiers journaux est saturé, procédez comme suit:

- Le gestionnaire de files d'attente reconnaît cette condition uniquement lors de la création d'un nouveau fichier journal de la taille requise, ce qu'il fait avant le moment où il est nécessaire.

- Il détecte la condition de disque saturé lorsque le système d'exploitation renvoie une erreur de la demande d'extension du fichier à la taille requise.
- Le gestionnaire de files d'attente émet le message AMQ6708 dans le journal des erreurs du gestionnaire de files d'attente.
- Un enregistrement First Failure Support Technology (FFST) est écrit dans le répertoire des erreurs à l'échelle du système. Cet enregistrement fournit des détails sur la condition de saturation du disque et doit être conservé si vous devez contacter le support IBM .

Les fichiers journaux sont créés à leur taille fixe, au lieu d'être étendus au fur et à mesure que les enregistrements de journal y sont écrits. Cela signifie que IBM MQ peut être à court d'espace disque uniquement lorsqu'il crée un nouveau fichier ; il ne peut pas être à court d'espace lorsqu'il écrit un enregistrement dans le journal. IBM MQ sait toujours combien d'espace est disponible dans les fichiers journaux existants et gère l'espace dans les fichiers en conséquence.

Lorsque vous utilisez la journalisation linéaire, vous avez la possibilité d'utiliser:

- Gestion automatique des extensions de journal.

Pour plus d'informations sur les nouveaux attributs de journal, voir [DISPLAY QMSTATUS](#) .

Consultez également les commandes suivantes ou leurs équivalents PCF:

- [RESET QMGR](#)
- [SET LOG](#) pour les plateformes réparties

- Options contrôlant l'utilisation des images de support.

Voir la commande [ALTER QMGR](#) et [ALTER QUEUES](#) pour plus d'informations sur:

- [IMGINTVL](#)
- [IMGLOGLN](#)
- [IMGRCOVO](#)
- [IMGRCOVQ](#)
- [IMGSCHEM](#)

La journalisation avec réutilisation automatique des journaux renvoie un problème de ressource.

Si vous êtes toujours à court d'espace, vérifiez que la configuration du journal dans le fichier de configuration du gestionnaire de files d'attente est correcte. Vous pouvez réduire le nombre de fichiers journaux principaux ou secondaires afin que le journal ne dépasse pas l'espace disponible.

Vous ne pouvez pas modifier la taille des fichiers journaux d'un gestionnaire de files d'attente existant. Le gestionnaire de files d'attente requiert que toutes les extensions de journal aient la même taille.

Gestion des fichiers journaux

Allouez suffisamment d'espace pour vos fichiers journaux. Pour la journalisation linéaire, vous pouvez supprimer les anciens fichiers journaux lorsqu'ils ne sont plus nécessaires.

Informations spécifiques à la journalisation avec réutilisation automatique des journaux

Si vous utilisez la journalisation avec réutilisation automatique des journaux, vérifiez qu'il y a suffisamment d'espace pour stocker les fichiers journaux lorsque vous configurez votre système (voir «Section LogDefaults du fichier mqs.ini», à la page 106 et «Section de journal du fichier qm.ini», à la page 147). La quantité d'espace disque utilisée par le journal n'augmente pas au-delà de la taille configurée, y compris l'espace pour les fichiers secondaires à créer lorsque cela est nécessaire.

Informations spécifiques à la journalisation linéaire

Si vous utilisez un journal linéaire, les fichiers journaux sont ajoutés en permanence à mesure que les données sont consignées et la quantité d'espace disque utilisée augmente avec le temps. Si le débit des données consignées est élevé, l'espace disque est rapidement utilisé par les nouveaux fichiers journaux.

Au fil du temps, les anciens fichiers journaux d'un journal linéaire ne sont plus nécessaires pour redémarrer le gestionnaire de files d'attente ou pour effectuer une reprise sur support de tout objet endommagé. Les méthodes suivantes déterminent les fichiers journaux qui sont encore requis:

Messages d'événement du consignateur

Lorsqu'un événement significatif se produit, par exemple une image de support d'enregistrement, des messages d'événement de consignateur sont générés. Le contenu des messages d'événement du consignateur indique les fichiers journaux qui sont encore requis pour le redémarrage du gestionnaire de files d'attente et la reprise sur incident lié au support. Pour plus d'informations sur les messages d'événement du consignateur, voir [Événements du consignateur](#)

Etat de gestionnaire de files d'attente

L'exécution de la commande MQSC, DISPLAY QMSTATUS ou de la commande PCF, Inquire Queue Manager Status, renvoie des informations sur le gestionnaire de files d'attente, y compris les détails des fichiers journaux requis. Pour plus d'informations sur les commandes MQSC, voir [Administration de IBM MQ à l'aide de commandes MQSC](#) et pour plus d'informations sur les commandes PCF, voir [Automatisation des tâches d'administration](#).

Messages du gestionnaire de files d'attente

Régulièrement, le gestionnaire de files d'attente émet une paire de messages pour indiquer quels fichiers journaux sont nécessaires:

- Le message AMQ7467I indique le nom du fichier journal le plus ancien requis pour redémarrer le gestionnaire de files d'attente. Ce fichier journal et tous les fichiers journaux plus récents doivent être disponibles lors du redémarrage du gestionnaire de files d'attente.
- Le message AMQ7468I indique le nom du fichier journal le plus ancien nécessaire à la reprise sur incident lié au support.

Pour déterminer les fichiers journaux "plus anciens" et "plus récents", utilisez le numéro de fichier journal plutôt que les heures de modification appliquées par le système de fichiers.

Informations applicables aux deux types de journalisation

Seuls les fichiers journaux requis pour le redémarrage du gestionnaire de files d'attente et les fichiers journaux actifs doivent être en ligne. Les fichiers journaux inactifs peuvent être copiés sur un support d'archivage, tel qu'une bande pour la reprise après incident, et supprimés du répertoire des journaux. Les fichiers journaux inactifs qui ne sont pas requis pour la récupération de support peuvent être considérés comme des fichiers journaux superflus. Vous pouvez supprimer les fichiers journaux superflus s'ils ne présentent plus d'intérêt pour votre opération.

Si un fichier journal nécessaire est introuvable, le message de l'opérateur AMQ6767E est émis. Mettez le fichier journal et tous les fichiers journaux suivants à la disposition du gestionnaire de files d'attente, puis renouvelez l'opération.

Nettoyage automatique des extensions de journal-consignation linéaire uniquement



Vous avez la possibilité d'utiliser la gestion automatique des extensions de journal linéaire qui ne sont plus nécessaires pour la reprise.

Vous utilisez l'attribut **LogManagement** dans la strophe Log du fichier qm.ini, ou à l'aide de IBM MQ Explorer, pour configurer la gestion automatique. Pour plus d'informations, voir [«Section de journal du fichier qm.ini»](#), à la page 147.

Voir le paramètre **LOG** de **DISPLAY QMSTATUS** pour plus de détails sur le fonctionnement du journal et les commandes suivantes pour l'utilisation du journal:

- [RESET QMGR](#)
- [Définir un journal](#)

Prise automatique d'images de support-consignation linéaire uniquement

Il existe un commutateur global pour contrôler si le gestionnaire de files d'attente écrit automatiquement des images de support, la valeur par défaut étant que le commutateur n'a pas été défini.

Vous pouvez contrôler si l'imagerie de support automatique se produit et la fréquence du processus, à l'aide des attributs de gestionnaire de files d'attente suivants:

IMGSCHED

Indique si le gestionnaire de files d'attente écrit automatiquement les images de support

IMGINTVL

Fréquence d'écriture des images de support, en minutes

IMGLOGLN

Mégaoctets de journal écrits depuis l'image de support précédente d'un objet.

Si vous avez un moment critique au cours de la journée où la charge de travail est très importante et que vous souhaitez vous assurer que le débit du système n'est pas impacté par la prise d'images de support automatiques, vous pouvez désactiver temporairement l'imagerie de support automatique en définissant **IMGSCHED(MANUAL)**.

Vous pouvez changer de **IMGSCHED** à tout moment au cours de la charge de travail.



Avertissement : MEDIALOG n'étant pas déplacé si vous ne prenez pas d'images de support, vous devez soit archiver les extensions, soit vous assurer que vous disposez d'un espace disque suffisant.

Vous pouvez également contrôler les images de support automatiques et manuelles pour d'autres objets définis par l'utilisateur à l'aide de l'attribut **IMGRCOVO** :

- Informations d'authentification
- Canal
- Connexion client
- Programme d'écoute
- Liste de noms
- Processus
- File d'attente d'alias
- File d'attente locale
- Service
- Topic

Pour les objets système internes, tels que le catalogue d'objets et l'objet gestionnaire de files d'attente, le gestionnaire de files d'attente écrit automatiquement les images de support appropriées.

Pour plus d'informations sur les attributs, voir [ALTER QMGR](#) .

Vous pouvez également activer ou désactiver les images de support automatiques et manuelles pour les files d'attente dynamiques locales et permanentes uniquement. Pour ce faire, utilisez l'attribut de file d'attente **IMGRCOVQ** .

Pour plus d'informations sur l'attribut **IMGRCOVQ** , voir [ALTER QUEUES](#) .

Remarques :

1. Les images de support sont prises en charge uniquement si vous utilisez la journalisation linéaire. Si vous avez activé les images de support automatique, mais que vous utilisez la journalisation avec réutilisation automatique des journaux, un message d'erreur est émis et l'attribut d'images de support automatique du gestionnaire de files d'attente est désactivé.

2. Si vous avez activé les images de support automatiques, mais que vous n'avez pas indiqué de fréquence, en minutes ou en mégaoctets, un message d'erreur est émis et aucune image de support automatique n'est écrite.
3. Vous pouvez enregistrer manuellement une image de support à l'aide de la commande `rcdmqimg` lorsque vous avez défini **IMGSCHED(AUTO)**, si vous le souhaitez.

Cela vous permet de prendre des images de support à un moment qui convient à votre entreprise, par exemple, lorsque votre système est silencieux. L'imagerie de support automatique tient compte de ces images de support manuelles, car la prise d'une image de support manuelle réinitialise l'intervalle et la longueur du journal, avant lesquels l'image de support automatique suivante est prise.

4. Le gestionnaire de files d'attente écrit les messages persistants uniquement dans les images de support et non dans les messages non persistants. Cela peut réduire la taille des images de support lors de la migration vers des versions IBM MQ ultérieures.

Comment définir **IMGLOGLN** et **IMGINTVL**

V 9.4.0 Par défaut, **IMGLOGLN** est défini sur `off` pour les gestionnaires de files d'attente autres que les gestionnaires de files d'attente Native HA. (Les gestionnaires de files d'attente Native HA sont créés avec **IMGLOGLN** défini sur la valeur de 25% de l'espace disponible sur le volume où les journaux de reprise doivent être écrits.)

V 9.4.0 Par défaut, **IMGINTVL** est défini sur 60 minutes. L'intervalle spécifié par **IMGINTVL** est respecté lorsqu'un nombre suffisant de nouveaux travaux a été effectué sur le gestionnaire de files d'attente pour qu'il soit intéressant d'enregistrer une nouvelle image. Sinon, la prise de nouvelles images est retardée.

Vous pouvez modifier les valeurs de **IMGLOGLN** et **IMGINTVL** pour obtenir la meilleure solution pour votre configuration. Rendez **IMGLOGLN** et **IMGINTVL** assez grands, de sorte que le gestionnaire de files d'attente ne passe qu'une fraction de son temps à enregistrer des images de support, mais assez petits pour que:

- Les objets endommagés peuvent être récupérés dans un délai raisonnable, et
- Suffisamment petit pour que votre journal s'adapte à votre disque sans manque d'espace.

Si vous définissez **IMGLOGLN**, une bonne pratique consiste à rendre **IMGLOGLN** plusieurs fois la quantité de données dans vos files d'attente et plusieurs fois le débit de données de votre charge de travail. Plus **IMGLOGLN** est grand, moins votre gestionnaire de files d'attente passe de temps à enregistrer des images de support.

De même, si vous définissez **IMGINTVL**, il est recommandé de rendre **IMGINTVL** plusieurs fois le temps nécessaire au gestionnaire de files d'attente pour enregistrer une image de support. Vous pouvez déterminer le temps nécessaire à l'enregistrement d'une image de support en enregistrant une image manuellement.

Si **IMGLOGLN** et **IMGINTVL** sont trop grands, la récupération d'un objet endommagé peut prendre beaucoup de temps, car toutes les extensions depuis la dernière image de support doivent être relues.

Faites en sorte que **IMGLOGLN** et **IMGINTVL** soient suffisamment petits pour que le temps maximal nécessaire à la récupération d'un objet endommagé soit acceptable pour vous.

Si **IMGLOGLN** et **IMGINTVL** sont très grands, cela signifie que le journal devient très volumineux car les images de support sont enregistrées si rarement.



Avertissement : Assurez-vous qu'un journal de cette taille s'adapte confortablement à votre système de fichiers de journal, car votre charge de travail sera annulée si le système de fichiers de journal se remplit complètement.

Vous pouvez définir à la fois **IMGINTVL** et **IMGLOGLN**. Cela peut s'avérer utile pour s'assurer que les images de support automatiques sont prises régulièrement pendant une charge de travail importante (contrôlée par **IMGLOGLN**), mais qu'elles sont toujours prises occasionnellement lorsque la charge de travail est très légère (contrôlée par **IMGINTVL**).

IMGINTVL et **IMGLOGLN** sont les cibles de l'intervalle et de la longueur des données de journal entre lesquelles les images de support automatiques sont prises.

Ces attributs ne doivent pas être considérés comme un maximum ou un minimum fixe. En fait, le gestionnaire de files d'attente peut décider de planifier une image de support automatique plus tôt, si le gestionnaire de files d'attente s'aperçoit qu'il s'agit d'un bon moment:

- Etant donné que la file d'attente est vide, la prise de l'image de support est la plus efficace en termes de performances, et
- Une image de support n'a pas été enregistrée depuis un certain temps

Il peut arriver que l'écart entre les images de support automatiques soit un peu plus long que celui de **IMGINTVL** et / ou de **IMGLOGLN**.

L'écart entre les images de support peut être supérieur à **IMGLOGLN** si la quantité de données dans les files d'attente approche **IMGLOGLN**. L'écart entre les images de support peut être plus grand que **IMGINTVL** si l'enregistrement d'une image de support prend presque autant de temps que **IMGINTVL**.

Il s'agit d'une mauvaise pratique car le gestionnaire de files d'attente passe une grande partie de son temps à enregistrer des images de support.

Lors de l'utilisation de l'enregistrement automatique d'images de support, le gestionnaire de files d'attente enregistre une image de support pour chaque objet et chaque file d'attente individuellement, de sorte que le gestionnaire de files d'attente effectue le suivi de l'intervalle et de la longueur de journal entre les images séparément pour chaque objet.

Progressivement au fil du temps, l'enregistrement des images des médias est décalé, au lieu d'enregistrer des images des médias pour tous les objets en même temps. Cet étalement répartit l'impact sur les performances des images des supports d'enregistrement, et constitue un autre avantage de l'utilisation de l'enregistrement automatique des images des supports par rapport à l'enregistrement manuel.

Prise manuelle d'images de support-consignation linéaire uniquement

L'enregistrement d'une image de support d'une file d'attente implique l'écriture de tous les messages persistants de cette file d'attente dans le journal. Pour les files d'attente contenant de grands volumes de données de message, cela implique l'écriture d'une grande quantité de données dans le journal, et ce processus peut avoir un impact sur les performances du système pendant qu'il se produit.

L'enregistrement d'images de support d'autres objets est probablement relativement rapide, car l'image de support d'autres objets ne contient pas de données utilisateur.

Vous devez déterminer avec soin quand enregistrer les images de support des files d'attente, afin que le processus n'interfère pas avec votre charge de travail maximale.

Vous devez enregistrer régulièrement l'image de support de tous les objets afin de mettre à jour l'extension de journal la plus ancienne nécessaire à la reprise sur incident.

Le moment est venu d'enregistrer l'image de support d'une file d'attente lorsqu'elle est vide, car à ce stade, aucune donnée de message n'est écrite dans le journal. A l'inverse, un mauvais moment se produit lorsque la file d'attente est très profonde ou qu'elle contient des messages très volumineux.

Un bon moment pour enregistrer l'image de support d'une file d'attente est lorsque votre système est silencieux, alors qu'un mauvais moment est pendant les pics de charge de travail. Si votre charge de travail est toujours silencieuse à minuit, par exemple, vous pouvez décider d'enregistrer des images de support à minuit chaque nuit.

L'échelonnement de l'enregistrement de chacune de vos files d'attente peut étendre l'impact sur les performances et donc réduire son effet. Plus il est long depuis que vous avez enregistré les dernières images de support, plus il est important de les enregistrer, car le nombre d'extensions de journal requises pour la reprise sur incident est en augmentation.

Remarque : Lors de la reprise sur incident lié au support, tous les fichiers journaux requis doivent être disponibles simultanément dans le répertoire des fichiers journaux. Veillez à prendre régulièrement des

images de support des objets que vous souhaitez peut-être récupérer afin d'éviter un manque d'espace disque pour stocker tous les fichiers journaux requis.

Par exemple, pour prendre une image de support de tous les objets de votre gestionnaire de files d'attente, exécutez la commande **rcdmqimg** comme illustré dans les exemples suivants:

Windows **Sous Windows**

```
rcdmqimg -m QMNAME -t all *
```

Linux **Sous AIX and Linux**

```
rcdmqimg -m QMNAME -t all "*"
```

L'exécution de **rcdmqimg** déplace le numéro de séquence du journal des supports (LSN) vers l'avant. Pour plus de détails sur les numéros de séquence de journal, voir [«Vidage du contenu du journal à l'aide de la commande dmpmqlog»](#), à la page 733. **rcdmqimg** ne s'exécute pas automatiquement ; par conséquent, il doit être exécuté manuellement ou à partir d'une tâche automatique que vous avez créée. Pour plus d'informations sur cette commande, voir [rcdmqimg](#) et [dmpmqlog](#).

L'enregistrement manuel des images de support avec **rcdmqimg** pour gérer l'espace de journalisation n'est pas nécessaire si vous avez choisi d'utiliser la journalisation linéaire avec l'imagerie de support automatique contrôlée par le gestionnaire de files d'attente.

Remarque : Les messages AMQ7467 et AMQ7468 peuvent également être émis lors de l'exécution de la commande **rcdmqimg**.

Images de support partielles

Il est recommandé d'utiliser les messages IBM MQ uniquement pour les données qui doivent être consommées dans un avenir proche, de sorte que chaque message soit placé dans une file d'attente pendant une durée relativement courte.

A l'inverse, il est peu pratique d'utiliser des messages IBM MQ pour stocker des données à long terme comme une base de données.

Il est également recommandé de s'assurer que vos files d'attente sont relativement peu profondes et que les files d'attente dont les messages sont en file d'attente depuis longtemps sont peu profondes.

En suivant ces instructions, vous activez le gestionnaire de files d'attente pour optimiser les performances de l'enregistrement automatique des images de support.

L'enregistrement de l'image de support d'une file d'attente vide est très efficace (du point de vue des performances), tandis que la prise de l'image de support d'une file d'attente contenant une grande quantité de données est très inefficace, car toutes ces données doivent être écrites dans le journal de l'image de support.

Pour les files d'attente peu profondes contenant des messages récemment insérés, le gestionnaire de files d'attente peut effectuer une optimisation supplémentaire.

Si tous les messages actuellement dans la file d'attente ont été insérés dans le passé récent, le gestionnaire de files d'attente peut être en mesure d'enregistrer l'image de support pour le compte d'une heure (*point de récupération*) juste avant que tous les messages ne soient insérés, et donc d'enregistrer l'image de la file d'attente vide. Ce procédé est très peu coûteux en termes de performances.

Si tous les messages qui se trouvaient dans la file d'attente au point de reprise ont été reçus par la suite, ces messages n'ont pas besoin d'être enregistrés dans l'image du support, car ils ne se trouvent plus dans la file d'attente.

Il s'agit d'une *image de support partielle*. Ensuite, dans le cas peu probable où la file d'attente doit être récupérée, tous les enregistrements de journaux liés à cette file d'attente depuis la dernière image de support seront réexécutés, ce qui permet de restaurer tous les messages récemment insérés.

Même s'il y avait quelques messages dans la file d'attente au point de reprise, qui sont actuellement dans la file d'attente (et doivent donc être enregistrés dans l'image de support partiel), il est encore plus efficace d'enregistrer cette image de support partiel plus petite, qu'une image de support complet de tous les messages.

Le fait de s'assurer que les messages restent dans les files d'attente pendant une courte période est susceptible d'améliorer les performances de l'enregistrement automatique des images de support.

Emplacement de fichier journal

Lorsque vous choisissez un emplacement pour vos fichiers journaux, n'oubliez pas que l'opération est gravement affectée si IBM MQ ne parvient pas à formater un nouveau journal en raison d'un manque d'espace disque.

Si vous utilisez un journal circulaire, vérifiez qu'il y a suffisamment d'espace sur l'unité pour au moins les fichiers journaux principaux configurés. Laissez également de l'espace pour au moins un fichier journal secondaire, ce qui est nécessaire si le journal doit augmenter.

Si vous utilisez un journal linéaire, accordez beaucoup plus d'espace ; l'espace consommé par le journal augmente continuellement à mesure que les données sont consignées.

Vous devez placer les fichiers journaux sur une unité de disque distincte des données du gestionnaire de files d'attente.

L'intégrité des données sur ce périphérique est primordiale-vous devez autoriser la redondance intégrée.

Il peut également être possible de placer les fichiers journaux sur plusieurs unités de disque en miroir. Cela protège contre les pannes de l'unité contenant le journal. Sans la mise en miroir, vous pourriez être obligé de revenir à la dernière sauvegarde de votre système IBM MQ .

Remplacement du journal linéaire par le journal circulaire de votre gestionnaire de files d'attente

Vous pouvez modifier le type de journal de votre gestionnaire de files d'attente en le faisant passer de linéaire à circulaire à l'aide de la commande **migmqlog** .

Avant de commencer

Déterminez si vous souhaitez utiliser une journalisation linéaire ou circulaire en consultant la rubrique [Types de journalisation](#).

Indiquez si vous souhaitez modifier le type de journal ou déplacer le journal vers un nouvel emplacement. Lorsque vous déplacez le journal vers un nouvel emplacement à l'aide de la commande **migmqlog** , le chemin d'accès au journal dans le fichier `qm.ini` est mis à jour de sorte que lorsque vous démarrez votre gestionnaire de files d'attente, il utilise le journal modifié. Vous pouvez spécifier un nouvel emplacement à l'aide de l'option **-ld** . Si vous déplacez votre gestionnaire de files d'attente d'un ancien disque vers un nouveau disque au format avancé, il est recommandé d'utiliser l'option **-ld**.

Procédure

1. Connectez-vous en tant que membre du groupe `mqm`.
2. Vérifiez que vous disposez de suffisamment d'espace pour modifier votre journal. Vous devez vous assurer qu'il y a de l'espace pour au moins les fichiers journaux principaux configurés et un fichier journal secondaire.
3. Si ce n'est pas déjà fait, arrêtez le gestionnaire de files d'attente à l'aide de la commande **endmqm -w** .
4. Si ce n'est pas déjà fait, effectuez une sauvegarde de votre gestionnaire de files d'attente.
Pour plus d'informations, voir «[Sauvegarde des données de gestionnaire de files d'attente](#)», à la page 738 et «[Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente](#)», à la page 15.
5. Exécutez la commande **migmqlog** :

- Si vous choisissez de modifier votre type de journal sans modifier l'emplacement du journal, utilisez la commande suivante:

```
migmqlog -m QMgrName -lc
```

- Si vous choisissez de modifier votre type de journal et de déplacer le journal vers un nouvel emplacement, utilisez la commande suivante:

```
migmqlog -m QMgrName -lc -ld NewLogLocation
```

où *NewLogLocation* est un chemin d'accès absolu qui indique le nouvel emplacement du fichier journal. N'utilisez pas de chemin de fichier relatif avec le paramètre **-ld**.

Pour plus d'informations, voir **migmqlog**.

Si, pour une raison quelconque, par exemple en raison d'une panne de courant, la commande **migmqlog** s'arrête avant d'avoir terminé le processus, réexécutez la même commande **migmqlog** sur les journaux partiellement modifiés pour effectuer les modifications.

Résultats

La commande s'exécute et le type de journal du gestionnaire de files d'attente est mis à jour. Notez que l'exécution de **migmqlog** peut prendre quelques minutes si votre journal est très volumineux. Toutefois, la commande génère régulièrement des messages de progression.

Tâches associées

«Remplacement du journal de votre gestionnaire de files d'attente circulaire par le journal linéaire», à la page 725

Vous pouvez modifier le type de journal de votre gestionnaire de files d'attente en le faisant passer du type circulaire au type linéaire à l'aide de la commande **migmqlog**.

ALW Remplacement du journal de votre gestionnaire de files d'attente circulaire par le journal linéaire

Vous pouvez modifier le type de journal de votre gestionnaire de files d'attente en le faisant passer du type circulaire au type linéaire à l'aide de la commande **migmqlog**.

Avant de commencer

Déterminez si vous souhaitez utiliser une journalisation linéaire ou circulaire en consultant la rubrique [Types de journalisation](#).

Indiquez si vous souhaitez modifier le type de journal ou déplacer le journal vers un nouvel emplacement. Lorsque vous déplacez le journal vers un nouvel emplacement à l'aide de la commande **migmqlog**, le chemin d'accès au journal dans le fichier *qm.ini* est mis à jour de sorte que lorsque vous démarrez votre gestionnaire de files d'attente, il utilise le journal modifié. Vous pouvez spécifier un nouvel emplacement à l'aide de l'option **-ld**. Si vous déplacez votre gestionnaire de files d'attente d'un ancien disque vers un nouveau disque au format avancé, il est recommandé d'utiliser l'option **-ld**.

Pourquoi et quand exécuter cette tâche



Avertissement : Une fois que vous avez modifié votre journal, aucune image de support n'a été enregistrée au démarrage du gestionnaire de files d'attente. Prévoyez la manière d'enregistrer des images de support, automatiquement en définissant les attributs suivants :

- IMGSCHEd
- IMGINTVL
- IMGLOGLN
- IMGRCOVO
- IMGRCOVQ

dans [ALTER QMGR](#), ou manuellement en exécutant périodiquement la commande **rcdmqimg**.

Procédure

1. Connectez-vous en tant que membre du groupe mqm.
2. Vérifiez que vous disposez de suffisamment d'espace pour modifier votre journal. L'espace utilisé par un journal linéaire augmente de façon continue au fur et à mesure que les données sont consignées.
3. Si ce n'est pas déjà fait, arrêtez le gestionnaire de files d'attente à l'aide de la commande **endmqm -w**.
4. Si ce n'est pas déjà fait, effectuez une sauvegarde de votre gestionnaire de files d'attente.
Pour plus d'informations, voir «Sauvegarde des données de gestionnaire de files d'attente», à la page 738 et «Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente», à la page 15.
5. Exécutez la commande **migmqlog**. Notez que l'exécution de **migmqlog** peut prendre quelques minutes si votre journal est très volumineux. Toutefois, la commande génère régulièrement des messages de progression.

- Si vous choisissez de modifier votre type de journal sans modifier l'emplacement du journal, utilisez la commande suivante:

```
migmqlog -m QMgrName -ll
```

- Si vous choisissez de modifier votre type de journal et de déplacer le journal vers un nouvel emplacement, utilisez la commande suivante:

```
migmqlog -m QMgrName -ll -ld NewLogLocation
```

où *NewLogLocation* est un chemin d'accès absolu qui indique le nouvel emplacement du fichier journal. N'utilisez pas de chemin de fichier relatif avec le paramètre **-ld**.

Pour plus d'informations, voir **migmqlog**.

Si, pour une raison quelconque, par exemple en raison d'une panne de courant, la commande **migmqlog** s'arrête avant d'avoir terminé le processus, réexécutez la même commande **migmqlog** sur les journaux partiellement modifiés pour effectuer les modifications.

6. Démarrez le gestionnaire de files d'attente et définissez les attributs de file d'attente et de récupération d'image appropriés pour votre environnement.
7. Déterminez à quel moment des images manuelles des objets récupérables doivent être enregistrées.

Tâches associées

«Remplacement du journal linéaire par le journal circulaire de votre gestionnaire de files d'attente», à la page 724

Vous pouvez modifier le type de journal de votre gestionnaire de files d'attente en le faisant passer de linéaire à circulaire à l'aide de la commande **migmqlog**.

Détermination des fichiers journaux superflus-consignation linéaire uniquement

Pour la journalisation avec réutilisation automatique des journaux, ne supprimez jamais les données du répertoire de journaux. Lors de la gestion des fichiers journaux linéaires, il est important de vérifier quels fichiers peuvent être supprimés ou archivés. Ces informations vous aideront à prendre cette décision.

N'utilisez pas les heures de modification du système de fichiers pour déterminer les fichiers journaux "plus anciens". Utilisez uniquement le numéro de fichier journal. L'utilisation des fichiers journaux par le gestionnaire de files d'attente suit des règles complexes, y compris la pré-allocation et le formatage des fichiers journaux avant qu'ils ne soient nécessaires. Vous pouvez voir des fichiers journaux avec des heures de modification qui seraient trompeuses si vous essayez d'utiliser ces heures pour déterminer l'âge relatif.

Pour déterminer le fichier journal le plus ancien nécessaire, vous pouvez utiliser trois emplacements:

- Commande DISPLAY QMSTATUS
- Messages d'événement du consignateur et, enfin
- Messages du journal des erreurs

Pour la commande DISPLAY QMSTATUS, pour déterminer l'extension de journal la plus ancienne requise pour:

- Redémarrez le gestionnaire de files d'attente en exécutant la commande DISPLAY QMSTATUS RECLOG.
- Effectuez une reprise sur incident lié au support, à l'aide de la commande DISPLAY QMSTATUS MEDIALOG.
- Déterminez le nom de la notification d'archivage en exécutant la commande DISPLAY QMSTATUS ARCHLOG.

Vous pouvez réduire le nombre d'extensions de journal secondaire lors de l'utilisation de la journalisation avec réutilisation automatique des journaux en exécutant la commande **RESET QMGR TYPE (REDUCELOG)**.

En général, un numéro de fichier journal inférieur implique un journal plus ancien. Sauf si vous avez une rotation de fichier journal très élevée, de l'ordre de 3000 fichiers journaux par jour pendant 10 ans, vous n'avez pas besoin de traiter le renvoi à la ligne à 9 999 999. Dans ce cas, vous pouvez archiver tout fichier journal dont le nombre est inférieur à la valeur RECLOG et vous pouvez supprimer tout fichier journal dont le nombre est inférieur à la valeur RECLOG et à la valeur MEDIALOG.



Avertissement : Le fichier journal est renvoyé à la ligne, de sorte que le nombre suivant après 9 999 999 est égal à zéro.

Coldstart: Que faire si des extensions de journal sont manquantes ou endommagées

Si votre entreprise perd tout ou partie des extensions de journal nécessaires pour la reprise par redémarrage, le gestionnaire de files d'attente ne pourra pas réexécuter le journal de reprise et ne pourra donc pas redémarrer. Si vous demandez à votre gestionnaire de files d'attente de redémarrer lorsque le journal de reprise est endommagé de quelque manière que ce soit, au détriment du maintien de l'intégrité des données, il est possible de le faire, bien que cela soit fortement déconseillé. Ce processus est appelé *démarrage à froid* d'un gestionnaire de files d'attente.

Important : Le démarrage à froid d'un gestionnaire de files d'attente ne doit être envisagé que dans des circonstances exceptionnelles et comporte des risques pour l'intégrité des données, comme décrit sur cette page. IBM suggère de régénérer un gestionnaire de files d'attente, de préférence au démarrage à froid, en réponse à des fichiers de données endommagés.

Si un démarrage à froid est requis pour des raisons opérationnelles, demandez à votre représentant du support IBM de vérifier la cause première du problème. Vous devez remplacer un gestionnaire de files d'attente démarré à froid par un gestionnaire de files d'attente reconstruit dès que possible.

Les effets du coldstart

Lors du démarrage à froid, le gestionnaire de files d'attente crée un journal de reprise vide et s'appuie sur les données des fichiers de file d'attente et d'autres fichiers objet dans leur état existant. Etant donné que les données des fichiers de file d'attente peuvent être incohérentes, les messages peuvent être perdus, dupliqués, endommagés ou incohérents.

Le gestionnaire de files d'attente stocke la configuration de tous les autres objets conservés dans le journal de reprise, ainsi que dans les fichiers d'objet. D'autres données d'état internes sont également enregistrées dans le journal de reprise. Par conséquent, lors du démarrage à froid, les données d'état internes sont réinitialisées et toutes ces autres données de configuration peuvent être inexactes.

Les effets du démarrage à froid sont imprévisibles et variés, vous devez donc éviter un démarrage à froid à moins qu'il ne soit absolument nécessaire. Après le démarrage à froid, les informations des fichiers de file d'attente et d'objet peuvent être si incohérentes que le gestionnaire de files d'attente ne redémarrera pas du tout.

Si le gestionnaire de files d'attente redémarre, il n'existe pas de moyen simple de déterminer sur quelles données de message ou sur quelle configuration il est possible de s'appuyer. De plus, après un démarrage à froid, les files d'attente peuvent être endommagées et devenir donc totalement inutilisables.

De plus, si vous pouvez extraire une file d'attente particulière ou la placer dans une file d'attente particulière, les messages qu'elle contient peuvent être endommagés, manquants ou dupliqués. Les transactions et les canaux peuvent être bloqués en attente de validation. Même si votre gestionnaire de files d'attente démarre correctement et que les files d'attente sont intactes, les effets imprévisibles de ce démarrage risquent de ne se produire que plus tard.

Que faire si vous avez besoin de coldstart

L'exécution d'un démarrage à froid ne doit pas être considérée comme une pratique opérationnelle standard, et IBM vous en dissuade fortement. Toutefois, si vous êtes dans une position où vous devez absolument démarrer un gestionnaire de files d'attente, contactez le [Prise en charge de IBM MQ](#).

Le processus de démarrage à froid d'un gestionnaire de files d'attente était beaucoup plus compliqué pour un gestionnaire de files d'attente linéaire qu'un gestionnaire circulaire. Dans IBM MQ 9.1.3, le processus coldstart a été très simplifié et n'implique plus de copier ou de renommer des extensions de journal.

Contactez le support IBM, qui vous donnera une clé que vous transmettez à la commande **strmqm** pour démarrer un gestionnaire de files d'attente.



Avvertissement : La commande coldstart comporte toujours les mêmes risques de perte de l'intégrité des données qu'un démarrage à froid manuel, et IBM vous décourage fortement de le faire.

Elimination des futurs démarrages à froid: une demande

La commande **strmqm** requiert une clé pour coldstart, car IBM MQ vous demande de contacter le support IBM MQ si vous avez besoin de coldstart, car IBM MQ est soucieux de comprendre comment vous êtes arrivé à cette situation.

Il est évident que le démarrage à froid est quelque chose qu'il vaut mieux éviter. IBM MQ a déployé des efforts considérables pour vous assurer que vous n'aurez pas besoin de démarrer à froid votre gestionnaire de files d'attente, et IBM est désireux de découvrir s'il existe d'autres mesures que le produit peut prendre pour éviter d'avoir à démarrer à froid.

Précautions à prendre pour éviter un démarrage à froid

La méthode de consignation par défaut lors de la création d'un gestionnaire de files d'attente est la consignation circulaire. Avec la journalisation avec réutilisation automatique des journaux, vous autorisez le gestionnaire de files d'attente à utiliser un nombre particulier d'extensions de journal principal et secondaire d'une taille donnée. Créez votre système de fichiers journaux suffisamment grand pour contenir toutes les extensions de journal principale et secondaire, et vous ne devriez jamais avoir à les administrer.

Vous pouvez également utiliser la journalisation linéaire plutôt que la journalisation circulaire. La consignation linéaire vous donne la possibilité supplémentaire de récupérer des files d'attente et d'autres objets, dans le cas improbable où ils sont endommagés. Mais par défaut, la journalisation linéaire vous oblige à supprimer les extensions de journal qui ne sont plus nécessaires pour le redémarrage ou la reprise sur support. Il s'agit de la gestion manuelle des journaux.

Lors de l'administration des extensions de journal de cette manière, il est possible de supprimer par inadvertance un trop grand nombre d'extensions de journal et, par conséquent, d'avoir à démarrer à froid. Pour atténuer ce risque, utilisez la gestion automatique des journaux, de sorte que le gestionnaire de files d'attente gère les extensions de journal en votre nom.

La meilleure pratique consiste à placer votre journal de reprise dans un système de fichiers de journal distinct qui ne contient que le journal de reprise. Si vous placez votre journal de récupération dans le même système de fichiers que le reste de votre gestionnaire de files d'attente, vous pouvez parfois constater que ce système de fichiers se remplit accidentellement, peut-être en raison de fichiers de file d'attente volumineux. Faites du répertoire de journaux du gestionnaire de files d'attente un système

de fichiers distinct ou spécifiez un autre système de fichiers journaux à l'aide de l'option de ligne de commande **-ld** de la commande **crtmqm**.

Si le système de fichiers contenant les fichiers de file d'attente se remplit, il se peut que vous ne puissiez pas les placer dans ces files d'attente, mais le gestionnaire de files d'attente continue de s'exécuter. Si le système de fichiers contenant le journal de reprise se remplit, le gestionnaire de files d'attente s'arrête brutalement et ne redémarre pas tant que vous n'avez pas libéré de l'espace.

Veillez à ne pas supprimer les extensions de journal nécessaires à la reprise par redémarrage, sinon vous devrez peut-être démarrer à froid. Parfois, il se peut que vous deviez démarrer à froid car le disque qui contient le journal de reprise a échoué. La meilleure pratique consiste à placer le journal de reprise sur un disque répliqué et à réduire ainsi le risque de panne du disque.

Le déplacement de vos messages et de votre configuration vers un nouveau gestionnaire de files d'attente de remplacement évite la possibilité de problèmes permanents avec un gestionnaire de files d'attente précédemment démarré.

Notez quels gestionnaires de files d'attente ont déjà été démarrés, même s'ils ont été démarrés il y a longtemps et ont été arrêtés, redémarrés et migrés entre-temps. Lorsque vous contactez le support IBM, indiquez si le gestionnaire de files d'attente a été précédemment démarré et, si tel est le cas, indiquez autant d'informations que possible sur ce qui a provoqué l'exigence d'un démarrage à froid.

Utilisation du journal pour la reprise

Vous pouvez utiliser les informations des journaux pour vous aider à effectuer une reprise après incident.

Vos données peuvent être endommagées de plusieurs manières. IBM MQ vous aide à effectuer une reprise à partir des éléments suivants:

- Un objet de données endommagé
- Une coupure de courant dans le système
- Un incident de communication

Cette section explique comment les journaux sont utilisés pour résoudre ces problèmes.

Reprise après une coupure de courant ou des pannes de communication

IBM MQ peut être restauré suite à des pannes de communication et à une perte d'alimentation. Il peut également parfois être restauré à partir d'autres types de problèmes, tels que la suppression accidentelle d'un fichier.

En cas d'échec des communications, les messages persistants restent dans les files d'attente jusqu'à ce qu'ils soient supprimés par une application réceptrice. Si le message est transmis, il reste dans la file d'attente de transmission jusqu'à ce qu'il puisse être transmis avec succès. Pour effectuer une reprise après un incident de communication, vous pouvez généralement redémarrer les canaux à l'aide de la liaison qui a échoué.

Si vous perdez l'alimentation, lorsque le gestionnaire de files d'attente est redémarré, IBM MQ restaure les files d'attente à leur état validé au moment de l'échec. Cela garantit qu'aucun message persistant n'est perdu. Les messages non persistants sont supprimés ; ils ne survivent pas lorsque IBM MQ s'arrête brutalement.

Récupération des objets endommagés

Il existe des façons dont un objet IBM MQ peut devenir inutilisable, par exemple en raison de dommages accidentels. Vous devez ensuite récupérer l'intégralité de votre système ou une partie de celui-ci. L'action requise dépend du moment où le dommage est détecté, de la prise en charge de la reprise sur incident lié au support par la méthode de consignation sélectionnée et des objets endommagés.

Reprise sur incident lié au support

Vous pouvez enregistrer des images de support pour des objets afin qu'ils puissent être récupérés s'ils sont endommagés. Cette fonction est disponible uniquement sur les gestionnaires de files d'attente qui utilisent la journalisation linéaire ou la journalisation répliquée et, pour la journalisation linéaire,

uniquement pour les objets définis comme récupérables. Vous définissez que les types d'objet sont récupérables à l'aide des attributs de gestionnaire de files d'attente **IMGRCOVO** et **IMGRCOVQ**. Voir [ALTER QMGR](#). Si un objet qui n'est pas défini comme récupérable est endommagé, les options de reprise sont les mêmes que pour la journalisation avec réutilisation automatique des journaux.

La reprise sur incident lié au support recrée des objets à partir d'informations enregistrées dans un journal linéaire ou répliqué. Par exemple, si un fichier objet est supprimé par inadvertance ou devient inutilisable pour une autre raison, la reprise sur incident sur support peut le recréer. Les informations du journal requises pour la reprise sur incident lié au support d'un objet sont appelées *image de support*.

Une image de support est une séquence d'enregistrements de journal contenant une image d'un objet à partir de laquelle l'objet lui-même peut être recréé.

Le premier enregistrement de journal requis pour recréer un objet est appelé *enregistrement de reprise sur incident lié au support*; il s'agit du début de l'image de support la plus récente pour l'objet. L'enregistrement de reprise sur incident lié au support de chaque objet est l'une des informations enregistrées lors d'un point de contrôle.

Lorsqu'un objet est recréé à partir de son image de support, il est également nécessaire de relire tous les enregistrements de journal décrivant les mises à jour effectuées sur l'objet depuis la dernière image.

Prenons l'exemple d'une file d'attente locale dont l'image de l'objet file d'attente est prise avant qu'un message persistant ne soit inséré dans la file d'attente. Pour recréer la dernière image de l'objet, il est nécessaire de relire les entrées de journal enregistrant l'insertion du message dans la file d'attente, en plus de relire l'image elle-même.

Lorsqu'un objet est créé, les enregistrements de journal écrits contiennent suffisamment d'informations pour recréer complètement l'objet. Ces enregistrements constituent la première image de support de l'objet. Ensuite, à chaque arrêt, le gestionnaire de files d'attente enregistre automatiquement les images de support comme suit:

- Images de tous les objets de processus et de toutes les files d'attente qui ne sont pas locales
- Images de files d'attente locales vides

Les images de support peuvent également être enregistrées manuellement à l'aide de la commande **rcdmqimg**, décrite dans [rcdmqimg](#). Cette commande écrit une image de support de l'objet IBM MQ.

Le gestionnaire de files d'attente enregistre automatiquement les images de support si **IMGSCHED(AUTO)** est défini. Pour plus d'informations, voir [ALTER QMGR](#) pour plus d'informations sur **IMGINTVL** et **INGLOGLN**.

Lorsqu'une image de support a été écrite, seuls les journaux contenant l'image de support, ainsi que tous les journaux créés après cette période, sont requis pour recréer les objets endommagés. L'avantage de la création d'images de support dépend de facteurs tels que la quantité de mémoire disponible et la vitesse à laquelle les fichiers journaux sont créés.

Reprise à partir d'images de support


Un gestionnaire de files d'attente récupère automatiquement certains objets à partir de leur image de support lors du démarrage du gestionnaire de files d'attente. Elle récupère automatiquement une file d'attente si elle était impliquée dans une transaction incomplète lors du dernier arrêt du gestionnaire de files d'attente et qu'elle est endommagée ou endommagée lors du redémarrage.

Vous devez récupérer d'autres objets manuellement à l'aide de la commande **rcrmqobj**, qui réexécute les enregistrements du journal pour recréer l'objet IBM MQ. L'objet est recréé à partir de sa dernière image trouvée dans le journal, avec tous les événements de journal applicables entre le moment où l'image a été sauvegardée et le moment où la commande de recréation a été émise. Si un objet IBM MQ est endommagé, les seules actions valides pouvant être effectuées sont sa suppression ou sa recréation par cette méthode. Les messages non persistants ne peuvent pas être récupérés de cette manière.

Pour plus d'informations sur la commande **rcrmqobj**, voir [rcrmqobj](#).


Le fichier journal contenant l'enregistrement de reprise sur incident lié au support, ainsi que tous les fichiers journaux suivants, doivent être disponibles dans le répertoire des fichiers journaux lors de la

tentative de reprise sur incident lié au support d'un objet. Si un fichier requis est introuvable, le message d'opérateur AMQ6767 est émis et l'opération de récupération de support échoue. Si vous ne prenez pas d'images de support standard des objets que vous souhaitez recréer, il se peut que l'espace disque soit insuffisant pour contenir tous les fichiers journaux requis pour recréer un objet.

 Les gestionnaires de files d'attente Native HA utilisent la journalisation répliquée. Ces gestionnaires de files d'attente tentent la reprise automatique des objets éligibles lorsque des dommages sont détectés. Une fois démarrés, les gestionnaires de files d'attente Native HA, par défaut, tentent automatiquement une reprise asynchrone lorsque des dommages d'objet sont détectés. La reprise peut ne pas être immédiatement possible si, par exemple, l'objet est utilisé par une application ou si les extensions de journal requises pour la reprise sur support ne sont pas disponibles. Dans ces situations, le processus de reprise asynchrone effectue régulièrement des relances. Si le problème qui a empêché la récupération est résolu, l'objet sera récupéré lors de la prochaine tentative, ou l'objet peut être récupéré manuellement à l'aide de la commande **rcrmqobj**.

Les fichiers d'objet existants

Le gestionnaire de files d'attente stocke les attributs des objets définis dans **runmqsc** dans des fichiers sur disque. Ces fichiers objet se trouvent dans des sous-répertoires du répertoire de données du gestionnaire de files d'attente.

 Par exemple, sur les plateformes AIX and Linux, les canaux sont stockés dans `/var/mqm/qmgrs/qmgr/channel`.

Les données de ces fichiers objet correspondent à l'image de support des objets. Si ces fichiers objet sont supprimés ou endommagés, l'objet stocké dans ce fichier est endommagé. À l'aide d'un gestionnaire de files d'attente de journalisation linéaire, les objets endommagés peuvent être récupérés à partir du journal à l'aide de la commande **rcrmqobj**. Les gestionnaires de files d'attente de journalisation répliquée (Native HA) tentent automatiquement de récupérer les objets endommagés lorsqu'ils sont détectés.

La plupart des fichiers objet contiennent uniquement les attributs de l'objet, de sorte que les fichiers de canal contiennent les attributs des canaux. Voici les exceptions :

- Catalogue

Le catalogue d'objets catalogue tous les objets de tous les types et est stocké dans `qmanager/QMQOBJCAT`.

- Fichiers de synchronisation

Le fichier de synchronisation contient des données d'état internes associées à tous les canaux.

- Files d'attente

Les fichiers de file d'attente contiennent à la fois les messages de cette file d'attente et les attributs de cette file d'attente.

Notez qu'aucun objet catalogue ou fichier de synchronisation n'est exposé dans **runmqsc** ou IBM MQ Explorer.

Le catalogue et le gestionnaire de files d'attente peuvent être enregistrés, mais ne peuvent pas être récupérés. Si ces objets sont endommagés, le gestionnaire de files d'attente s'arrête de manière préemptive et ces objets sont récupérés automatiquement au redémarrage.

Les abonnements ne sont pas répertoriés dans les objets à enregistrer ou à récupérer, car les abonnements durables sont stockés dans une file d'attente système. Pour enregistrer ou récupérer des abonnements durables, enregistrez ou récupérez `SYSTEM.DURABLE.SUBSCRIBER.QUEUE` à la place.

Récupération des objets endommagés lors du démarrage

Si le gestionnaire de files d'attente détecte un objet endommagé lors du démarrage, l'action qu'il effectue dépend du type d'objet et de la configuration du gestionnaire de files d'attente pour la prise en charge de la reprise sur incident lié au support.

Si l'objet gestionnaire de files d'attente est endommagé, le gestionnaire de files d'attente ne peut pas démarrer à moins qu'il puisse récupérer l'objet. Si le gestionnaire de files d'attente est configuré avec un journal linéaire et prend donc en charge la reprise sur incident lié au support, IBM MQ tente automatiquement de recréer l'objet gestionnaire de files d'attente à partir de ses images de support. Si la méthode de consignation sélectionnée ne prend pas en charge la reprise sur support, vous pouvez soit restaurer une sauvegarde du gestionnaire de files d'attente, soit supprimer le gestionnaire de files d'attente.

Si des transactions étaient actives lorsque le gestionnaire de files d'attente s'est arrêté, les files d'attente locales contenant les messages persistants non validés ou insérés dans ces transactions sont également nécessaires pour démarrer le gestionnaire de files d'attente. Si l'une de ces files d'attente locales est endommagée et que le gestionnaire de files d'attente prend en charge la reprise sur incident lié au support, il tente automatiquement de les recréer à partir de leurs images de support. Si l'une des files d'attente ne peut pas être récupérée, IBM MQ ne peut pas démarrer.

Si des files d'attente locales endommagées contenant des messages non validés sont détectées au cours du processus de démarrage sur un gestionnaire de files d'attente qui ne prend pas en charge la reprise sur support, les files d'attente sont marquées comme étant des objets endommagés et les messages non validés qu'elles contiennent sont ignorés. Cette situation est due au fait qu'il n'est pas possible d'effectuer une reprise sur support des objets endommagés sur un tel gestionnaire de files d'attente et qu'il ne reste plus qu'à les supprimer. Le message AMQ7472 est émis pour signaler tout dommage.

Récupération d'objets endommagés à d'autres moments

La reprise sur support des objets est automatique uniquement lors du démarrage (autre que pour les gestionnaires de files d'attente Native HA, qui utilisent la reprise automatique par défaut). A d'autres moments, lorsque des dommages sont détectés, le message d'opérateur AMQ7472 est émis et la plupart des opérations utilisant l'objet échouent avec le code retour MQRC_OBJECT_DAMAGED. Si l'objet gestionnaire de files d'attente est endommagé à tout moment après le démarrage du gestionnaire de files d'attente, le gestionnaire de files d'attente effectue un arrêt préventif. Lorsqu'un objet a été endommagé, vous pouvez le supprimer ou, si le gestionnaire de files d'attente utilise un journal linéaire, tenter de le récupérer à partir de son image de support à l'aide de la commande **rcrmqobj** (voir [rcrmqobj](#) pour plus de détails).


Si une file d'attente (ou un autre objet) est endommagée, **MEDIALOG** n'avance pas. En effet, **MEDIALOG** est le domaine le plus ancien requis pour la reprise sur incident lié au support. Si votre charge de travail continue, **CURRLOG** continuera d'avancer et de nouvelles extensions seront écrites. Selon votre configuration (y compris votre paramètre **LogManagement**), cela peut commencer à remplir votre système de fichiers journaux. Si le système de fichiers journaux se remplit complètement, les transactions sont annulées et le gestionnaire de files d'attente risque de s'arrêter brutalement. Par conséquent, lorsqu'une file d'attente est endommagée, il se peut que vous n'ayez qu'un temps d'action limité avant que votre gestionnaire de files d'attente ne s'arrête. Le temps dont vous disposez dépend de la fréquence à laquelle votre charge de travail provoque l'écriture de nouvelles extensions par le gestionnaire de files d'attente et de la quantité d'espace disponible dans votre système de fichiers journaux.

Si vous utilisez la gestion manuelle des journaux, il se peut que vous archiviez des extensions qui ne sont pas nécessaires pour la reprise par redémarrage, puis que vous les supprimiez du système de fichiers du journal, même si elles sont toujours nécessaires pour la reprise sur support. Cela est acceptable tant que vous pouvez les restaurer à partir de votre archive si nécessaire. Cette règle n'entraîne pas le remplissage de votre système de fichiers journaux lorsqu'une file d'attente est endommagée et que **MEDIALOG** cesse de progresser. Toutefois, si vous archivez et supprimez uniquement les domaines qui ne sont pas nécessaires pour le redémarrage ou la récupération de support, votre système de fichiers journal commence à se remplir si une file d'attente est endommagée.

Si vous utilisez la gestion automatique ou la gestion des journaux d'archivage, le gestionnaire de files d'attente ne réutilise pas les domaines qui sont encore nécessaires pour la reprise sur incident lié au support, même si vous les avez archivés et que vous avez notifié le gestionnaire de files d'attente à l'aide de la commande [SET LOG ARCHIVÉ](#). Par conséquent, si une file d'attente est endommagée, votre système de fichiers journaux commencera à se remplir.

Si une file d'attente est endommagée, les FFDC OBJECT DAMAGED sont écrits et **MEDIALOG** s'arrête d'avancer. L'objet endommagé peut être identifié à partir de l'outil de diagnostic de premier niveau (FFDC) ou parce qu'il s'agit de l'objet avec le **MEDIALOG** le plus ancien lorsque vous affichez son statut dans **runmqsc**.

Si votre système de fichiers journaux est en cours de remplissage et que vous êtes préoccupé par le fait que votre charge de travail soit annulée car le système de fichiers journaux est en cours de saturation, la récupération de l'objet ou la mise au repos de votre charge de travail peut arrêter ce processus.

 Dans le cas des gestionnaires de files d'attente Native HA qui utilisent la journalisation répliquée, la reprise automatique des objets endommagés est tentée. Une fois démarrés, les gestionnaires de files d'attente Native HA, par défaut, tentent automatiquement une reprise asynchrone lorsque des dommages d'objet sont détectés. La reprise peut ne pas être immédiatement possible si, par exemple, l'objet est utilisé par une application ou si les extensions de journal requises pour la reprise sur support ne sont pas disponibles. Dans ces situations, le processus de reprise asynchrone effectuée régulièrement des relances. Si le problème qui a empêché la récupération est résolu, l'objet sera récupéré lors de la prochaine tentative, ou l'objet peut être récupéré manuellement à l'aide de la commande **rcrmqobj**.

Protection des fichiers journaux IBM MQ

Ne touchez pas les fichiers journaux lorsqu'un gestionnaire de files d'attente est en cours d'exécution, la récupération peut être impossible. Utilisez les droits de superutilisateur ou mqm pour protéger les fichiers journaux contre les modifications par inadvertance.

Ne supprimez pas les fichiers journaux actifs manuellement lorsqu'un gestionnaire de files d'attente IBM MQ est en cours d'exécution. Si un utilisateur supprime par inadvertance les fichiers journaux qu'un gestionnaire de files d'attente doit redémarrer, IBM MQ **ne génère** aucune erreur et continue de traiter les données *y compris les messages persistants*. Le gestionnaire de files d'attente s'arrête normalement, mais le redémarrage peut échouer. La récupération des messages devient alors impossible.

Les utilisateurs ayant le droit de supprimer des journaux utilisés par un gestionnaire de files d'attente actif ont également le droit de supprimer d'autres ressources importantes du gestionnaire de files d'attente (telles que les fichiers de file d'attente, le catalogue d'objets et les fichiers exécutables IBM MQ). Ils peuvent donc endommager, peut-être par inexpérience, un gestionnaire de files d'attente en cours d'exécution ou en sommeil d'une manière contre laquelle IBM MQ ne peut pas se protéger.

Soyez prudent lorsque vous attribuez des droits de superutilisateur ou mqm.

Vidage du contenu du journal à l'aide de la commande dmpmqlog

Comment utiliser la commande `dmpmqlog` pour vider le contenu du journal du gestionnaire de files d'attente.

Utilisez la commande `dmpmqlog` pour vider le contenu du journal du gestionnaire de files d'attente. Par défaut, tous les enregistrements de journal actifs sont vidés, c'est-à-dire que la commande commence le vidage à partir de la tête du journal (généralement le début du dernier point de contrôle terminé).

Le journal peut généralement être vidé uniquement lorsque le gestionnaire de files d'attente n'est pas en cours d'exécution. Étant donné que le gestionnaire de files d'attente prend un point de contrôle lors de l'arrêt, la partie active du journal contient généralement un petit nombre d'enregistrements de journal. Toutefois, vous pouvez utiliser la commande `dmpmqlog` pour vider plusieurs enregistrements de journal à l'aide de l'une des options suivantes pour modifier la position de début du vidage:

- Démarrez le vidage à partir de la *base* du journal. La base du journal est le premier enregistrement de journal dans le fichier journal qui contient la tête du journal. La quantité de données supplémentaires vidées dans ce cas dépend de l'emplacement de la tête du journal dans le fichier journal. S'il est proche du début du fichier journal, seule une petite quantité de données supplémentaires est vidée. Si la tête est vers la fin du fichier journal, beaucoup plus de données sont vidées.
- Indiquez la position de début du vidage en tant qu'enregistrement de journal individuel. Chaque enregistrement de journal est identifié par un *numéro de séquence de journal (LSN)* unique. Dans le

cas de la journalisation avec réutilisation automatique des journaux, cet enregistrement de journal de début ne peut pas être antérieur à la base du journal ; cette restriction ne s'applique pas aux journaux linéaires. Vous devrez peut-être rétablir les fichiers journaux inactifs avant d'exécuter la commande. Vous devez spécifier un numéro de séquence de journal valide, extrait de la sortie `dmpmqlog` précédente, comme position de début.

Par exemple, avec la consignation linéaire, vous pouvez spécifier le `nextlsn` à partir de votre dernière sortie `dmpmqlog`. `nextlsn` apparaît dans `Log File Header` et indique le numéro de séquence de journal du prochain enregistrement de journal à écrire. Utilisez cette position de début pour formater tous les enregistrements de journal écrits depuis le dernier vidage du journal.

- **Pour les journaux linéaires uniquement**, vous pouvez demander à `dmpmqlog` de démarrer le formatage des enregistrements de journal à partir d'un domaine de fichier journal donné. Dans ce cas, `dmpmqlog` s'attend à trouver ce fichier journal, et chaque fichier journal successif, dans le même répertoire que les fichiers journaux actifs. Cette option ne s'applique pas aux journaux circulaires, où `dmpmqlog` ne peut pas accéder aux enregistrements de journal antérieurs à la base du journal.

La sortie de la commande `dmpmqlog` est `Log File Header` et une série d'enregistrements de journal formatés. Le gestionnaire de files d'attente utilise plusieurs enregistrements de journal pour enregistrer les modifications apportées à ses données.

Certaines des informations formatées sont uniquement utiles en interne. La liste suivante répertorie les enregistrements de journal les plus utiles:

En-tête du fichier journal

Chaque journal possède un en-tête de fichier journal unique, qui est toujours le premier élément formaté par la commande `dmpmqlog`. Il inclut les zones suivantes :

<i>logactive</i>	Nombre d'extensions de journal principal.
<i>loginactive</i>	Nombre d'extensions de journal secondaire.
<i>logsize</i>	Nombre de pages de 4 ko par domaine.
<i>baselsn</i>	Premier numéro de séquence de journal dans l'étendue du journal contenant la tête du journal.
<i>nextlsn</i>	Numéro de séquence de journal de l'enregistrement de journal suivant à écrire.
<i>headlsn</i>	Numéro de séquence de journal de l'enregistrement de journal en tête du journal.
<i>empennage</i>	Numéro de séquence de journal identifiant la position de la queue du journal.
<i>hflag1</i>	Indique si le journal est CIRCULAIRE ou LOG RETAIN (linéaire).
<i>IDHeadExtent</i>	Domaine réservé aux journaux contenant la tête du journal.

En-tête d'enregistrement de journal

Chaque enregistrement de journal dans le journal comporte un en-tête fixe contenant les informations suivantes:

<i>numéro de séquence de journal</i>	Numéro de séquence du journal.
<i>LogRecdType</i>	Type de l'enregistrement de journal.
<i>ID objet XTranid</i>	Identificateur de transaction associé à cet enregistrement de journal (le cas échéant).

La valeur *TranType* de MQI indique une transaction IBM MQuniquement. Un *TranType* de XA est impliqué avec d'autres gestionnaires de ressources. Les mises à jour impliquées dans la même unité d'oeuvre ont le même *XTranid*.

<i>QueueName</i>	File d'attente associée à cet enregistrement de journal (le cas échéant).
<i>QID</i>	Identificateur interne unique de la file d'attente.
<i>PrevLSN</i>	Numéro de séquence de journal de l'enregistrement de journal précédent dans la même transaction (le cas échéant).

Démarrer un gestionnaire de files d'attente

Ce journal indique que le gestionnaire de files d'attente a démarré.

<i>StartDate</i>	Date de démarrage du gestionnaire de files d'attente.
<i>StartTime</i>	Heure de démarrage du gestionnaire de files d'attente.

Arrêter un gestionnaire de files d'attente

Ce journal indique que le gestionnaire de files d'attente s'est arrêté.

<i>StopDate</i>	Date d'arrêt du gestionnaire de files d'attente.
<i>StopTime</i>	Heure à laquelle le gestionnaire de files d'attente s'est arrêté.
<i>ForceFlag</i>	Type d'arrêt utilisé.

DEBUT DU POINT DE CONTROLE

Indique le début d'un point de contrôle de gestionnaire de files d'attente.

FIN DU POINT DE CONTROLE

Indique la fin d'un point de contrôle de gestionnaire de files d'attente.

<i>ChkPtLSN</i>	Numéro de séquence de journal de l'enregistrement de journal qui a démarré ce point de contrôle.
-----------------	--

Insertion d'un message

Cette opération consigne un message persistant inséré dans une file d'attente. Si le message a été placé sous le point de synchronisation, l'en-tête de l'enregistrement de journal contient une valeur *XTranId* non nulle. Le reste de l'enregistrement contient:

<i>MapIndex</i>	Identificateur du message dans la file d'attente. Il peut être utilisé pour correspondre à la méthode MQGET correspondante qui a été utilisée pour extraire ce message de la file d'attente. Dans ce cas, un enregistrement de journal <i>Get Message</i> ultérieur peut être trouvé contenant les mêmes éléments <i>QueueName</i> et <i>MapIndex</i> . A ce stade, l'identificateur <i>MapIndex</i> peut être réutilisé pour un message d'insertion ultérieur dans cette file d'attente.
<i>Données</i>	Le vidage hexadécimal de cet enregistrement de journal contient diverses données internes, suivies d'une représentation du descripteur de message (eyecatcher MD), puis des données de message elles-mêmes.

Insertion d'une partie

Les messages persistants qui sont trop volumineux pour un enregistrement de journal unique sont consignés en tant que plusieurs enregistrements de journal *Put Part* suivis d'un enregistrement *Put Message* unique. S'il existe des enregistrements *Put Part*, la zone *PrevLSN* regroupe les enregistrements *Put Part* et l'enregistrement *Put Message* final.

<i>Données</i>	Poursuit le traitement des données de message lorsque l'enregistrement de journal précédent s'est arrêté.
----------------	---

Obtenir un message

Seules les extractions de messages persistants sont consignées. Si le message a été obtenu sous le point de synchronisation, l'en-tête de l'enregistrement de journal contient une valeur *XTranid* non nulle. Le reste de l'enregistrement contient:

<i>MapIndex</i>	Identifie le message extrait de la file d'attente. L'enregistrement de journal <i>Put Message</i> le plus récent contenant le même <i>QueueName</i> et <i>MapIndex</i> identifie le message qui a été extrait.
<i>Priorité QPriority</i>	Priorité du message extrait de la file d'attente.

Démarrer la transaction

Indique le début d'une nouvelle transaction. Un *TranType* de MQI indique une transaction IBM MQ uniquement. La valeur *TranType* de XA indique un type qui implique d'autres gestionnaires de ressources. Toutes les mises à jour effectuées par cette transaction auront le même *XTranid*.

Préparer la transaction

Indique que le gestionnaire de files d'attente est prêt à valider les mises à jour associées au *XTranid* spécifié. Cet enregistrement de journal est écrit dans le cadre d'une validation en deux phases impliquant d'autres gestionnaires de ressources.

Commit Transaction

Indique que le gestionnaire de files d'attente a validé toutes les mises à jour effectuées par une transaction.

Annuler la transaction

Indique que le gestionnaire de files d'attente a l'intention d'annuler une transaction.

Terminer la transaction

Indique la fin d'une transaction annulée.

Table de transactions

Cet enregistrement est écrit pendant le point de synchronisation. Il enregistre l'état de chaque transaction ayant effectué des mises à jour persistantes. Pour chaque transaction, les informations suivantes sont enregistrées:

<i>ID objet XTranid</i>	Identificateur de la transaction.
<i>FirstLSN</i>	Numéro de séquence de journal du premier enregistrement de journal associé à la transaction.
<i>LastLSN</i>	Numéro de séquence de journal du dernier enregistrement de journal associé à la transaction.

Participants à la transaction

Cet enregistrement de journal est écrit par le composant XA Transaction Manager du gestionnaire de files d'attente. Il enregistre les gestionnaires de ressources externes qui participent aux transactions. Pour chaque participant, les éléments suivants sont enregistrés:

<i>Nom de la commande RMName</i>	Nom du gestionnaire de ressources.
<i>ID de transaction</i>	Identificateur du gestionnaire de ressources. Il est également consigné dans les enregistrements de journal <i>Transaction Prepared</i> suivants qui enregistrent les transactions globales auxquelles participe le gestionnaire de ressources.
<i>SwitchFile</i>	Fichier de commutation de chargement de ce gestionnaire de ressources.
<i>XAOpenString</i>	Chaîne d'ouverture XA pour ce gestionnaire de ressources.
<i>XACloseString</i>	Chaîne de fermeture XA pour ce gestionnaire de ressources.

Transaction préparée

Cet enregistrement de journal est écrit par le composant XA Transaction Manager du gestionnaire de files d'attente. Indique que la transaction globale spécifiée a été correctement préparée. Chacun des gestionnaires de ressources participants sera invité à valider. Le *RMID* de chaque gestionnaire de ressources préparé est enregistré dans l'enregistrement de journal. Si le gestionnaire de files d'attente lui-même participe à la transaction, un *Participant Entry* avec un *RMID* égal à zéro est présent.

Effacement de la transaction

Cet enregistrement de journal est écrit par le composant XA Transaction Manager du gestionnaire de files d'attente. Il suit l'enregistrement de journal *Transaction Prepared* lorsque la décision de validation a été transmise à chaque participant.

purger la file d'attente

Cela consigne le fait que tous les messages d'une file d'attente ont été purgés, par exemple, à l'aide de la commande MQSC CLEAR QUEUE.

Attributs File d'attente

Cette opération consigne l'initialisation ou la modification des attributs d'une file d'attente.

Créer un objet

Cette opération consigne la création d'un objet IBM MQ .

<i>ObjName</i>	Nom de l'objet créé.
<i>UserId</i>	ID utilisateur effectuant la création.

Supprimer l'objet

Cette opération consigne la suppression d'un objet IBM MQ .

<i>ObjName</i>	Nom de l'objet qui a été supprimé.
----------------	------------------------------------

Sauvegarde et restauration des données du gestionnaire de files d'attente IBM MQ

Vous pouvez protéger les gestionnaires de files d'attente contre les éventuelles altérations causées par des pannes matérielles en sauvegardant les gestionnaires de files d'attente et les données de gestionnaire de files d'attente, en sauvegardant uniquement la configuration du gestionnaire de files d'attente et en utilisant un gestionnaire de files d'attente de sauvegarde.

Pourquoi et quand exécuter cette tâche



ATTENTION : Vous devez être très prudent si vous déplacez un gestionnaire de files d'attente vers un autre système d'exploitation. Pour plus d'informations, voir [Déplacement d'un gestionnaire de files d'attente vers un autre système d'exploitation](#) .

Régulièrement, vous pouvez prendre des mesures pour protéger les gestionnaires de files d'attente contre les éventuelles altérations causées par des pannes matérielles. Il existe trois façons de protéger un gestionnaire de files d'attente:

Sauvegarde des données du gestionnaire de files d'attente

Si le matériel tombe en panne, l'arrêt d'un gestionnaire de files d'attente peut être forcé. Si des données de journal de gestionnaire de files d'attente sont perdues en raison d'une panne matérielle, il se peut que le gestionnaire de files d'attente ne puisse pas redémarrer. Si vous sauvegardez les données du gestionnaire de files d'attente, vous pouvez récupérer certaines ou toutes les données perdues du gestionnaire de files d'attente.

En général, plus vous sauvegardez fréquemment les données du gestionnaire de files d'attente, moins vous perdez de données en cas de panne matérielle entraînant une perte d'intégrité du journal de reprise.

Pour sauvegarder les données du gestionnaire de files d'attente, le gestionnaire de files d'attente ne doit pas être en cours d'exécution.

Sauvegarde de la configuration du gestionnaire de files d'attente uniquement

Si le matériel tombe en panne, l'arrêt d'un gestionnaire de files d'attente peut être forcé. Si la configuration du gestionnaire de files d'attente et les données de journal sont perdues en raison d'une panne matérielle, le gestionnaire de files d'attente ne peut pas redémarrer ou être récupéré à partir du journal. Si vous sauvegardez la configuration du gestionnaire de files d'attente, vous pouvez recréer le gestionnaire de files d'attente et tous ses objets à partir de définitions sauvegardées.

Pour sauvegarder la configuration du gestionnaire de files d'attente, le gestionnaire de files d'attente doit être en cours d'exécution.

Utiliser un gestionnaire de files d'attente de sauvegarde

Si l'incident matériel est grave, il se peut qu'un gestionnaire de files d'attente soit irrécupérable. Dans cette situation, si le gestionnaire de files d'attente irrécupérable possède un gestionnaire de files d'attente de sauvegarde dédié, le gestionnaire de files d'attente de sauvegarde peut être activé à la place du gestionnaire de files d'attente irrécupérable. S'il est mis à jour régulièrement, le journal du gestionnaire de files d'attente de sauvegarde peut contenir des données de journal incluant le dernier journal complet du gestionnaire de files d'attente irrécupérable.

Un gestionnaire de files d'attente de sauvegarde peut être mis à jour alors que le gestionnaire de files d'attente existant est toujours en cours d'exécution.

Procédure

- Pour sauvegarder et restaurer les données du gestionnaire de files d'attente, voir:
 - [«Sauvegarde des données de gestionnaire de files d'attente»](#), à la page 738.
 - [«Restauration des données du gestionnaire de files d'attente»](#), à la page 739.
- Pour sauvegarder et restaurer la configuration du gestionnaire de files d'attente, voir:
 - [«Sauvegarde de la configuration du gestionnaire de files d'attente»](#), à la page 740
 - [«Restauration de la configuration du gestionnaire de files d'»](#), à la page 741
- Pour créer, mettre à jour et démarrer un gestionnaire de files d'attente de sauvegarde, voir [«Utilisation d'un gestionnaire de files d'attente de sauvegarde»](#), à la page 742.

Sauvegarde des données de gestionnaire de files d'attente

La sauvegarde des données du gestionnaire de files d'attente peut vous aider à vous protéger contre les pertes de données causées par des erreurs matérielles.

Avant de commencer

Avant de commencer à sauvegarder le gestionnaire de files d'attente, vérifiez qu'il n'est pas en cours d'exécution. Si vous tentez d'effectuer une sauvegarde d'un gestionnaire de files d'attente en cours d'exécution, la sauvegarde risque de ne pas être cohérente en raison des mises à jour en cours lors de la copie des fichiers. Si possible, arrêtez votre gestionnaire de files d'attente en exécutant la commande **endmqm -w** (arrêt en attente). Si cette opération échoue, utilisez la commande **endmqm -i** (arrêt immédiat).

Pourquoi et quand exécuter cette tâche

Pour effectuer une copie de sauvegarde des données d'un gestionnaire de files d'attente, procédez comme suit:

Procédure

1. Recherchez les répertoires dans lesquels le gestionnaire de files d'attente place ses données et ses fichiers journaux, à l'aide des informations des fichiers de configuration.

Pour plus d'informations, voir [«Modification des informations de configuration IBM MQ dans les fichiers .ini sur Multiplatforms»](#), à la page 96.

Remarque : Les noms qui apparaissent dans le répertoire sont transformés pour s'assurer qu'ils sont compatibles avec la plateforme sur laquelle vous utilisez IBM MQ. Pour plus d'informations sur les transformations de nom, voir [Présentation des noms de fichier IBM MQ](#).


2. Effectuez des copies de tous les répertoires de données et de fichiers journaux du gestionnaire de files d'attente, y compris de tous les sous-répertoires.

Veillez à ne pas manquer de fichiers, en particulier le fichier de contrôle de journal, comme décrit dans «[A quoi ressemblent les journaux](#)», à la page 703, et les fichiers de configuration, comme décrit dans «[Fichiers d'initialisation et de configuration](#)», à la page 264. Certains répertoires peuvent être vides, mais vous en aurez besoin pour restaurer la sauvegarde à une date ultérieure.

Pour la journalisation avec réutilisation automatique des journaux, sauvegardez simultanément les répertoires de données et de fichiers journaux du gestionnaire de files d'attente afin de pouvoir restaurer un ensemble cohérent de données et de journaux du gestionnaire de files d'attente.

Pour la journalisation linéaire, sauvegardez en même temps les répertoires des fichiers journaux et des données du gestionnaire de files d'attente. Il est possible de restaurer uniquement les fichiers de données du gestionnaire de files d'attente si une séquence complète correspondante de fichiers journaux est disponible.

3. Conservez les propriétés des fichiers.

 Pour les systèmes IBM MQ for UNIX et Linux, vous pouvez effectuer cette opération à l'aide de la commande **tar**. (Si vos files d'attente sont supérieures à 2 Go, vous ne pouvez pas utiliser la commande **tar**. Pour plus d'informations, voir [Activation des files d'attente volumineuses](#).)

Remarque : Lorsque vous effectuez une mise à niveau vers IBM WebSphere MQ 7.5 ou une version ultérieure, veillez à effectuer une sauvegarde du fichier `qm.ini` et des entrées de registre. Les informations du gestionnaire de files d'attente sont stockées dans le fichier `qm.ini` et peuvent être utilisées pour rétablir une version précédente d'IBM MQ.

Tâches associées

[Arrêt d'un gestionnaire de files d'attente](#)

«[Sauvegarde des fichiers de configuration après la création d'un gestionnaire de files d'attente](#)», à la page 15

Les informations de configuration IBM MQ sont stockées dans des fichiers de configuration sous AIX, Linux, and Windows. Après avoir créé un gestionnaire de files d'attente, sauvegardez vos fichiers de configuration. Ensuite, si vous créez un autre gestionnaire de files d'attente à l'origine des problèmes, vous pouvez rétablir les sauvegardes une fois que vous avez supprimé la source du problème.

Restauration des données du gestionnaire de files d'attente

Procédez comme suit pour restaurer une sauvegarde des données d'un gestionnaire de files d'attente.

Avant de commencer

Avant de démarrer la sauvegarde, vérifiez que le gestionnaire de files d'attente n'est pas en cours d'exécution.

Lors de la restauration d'une sauvegarde d'un gestionnaire de files d'attente dans un cluster, voir «[Récupération d'un gestionnaire de files d'attente de cluster](#)», à la page 400 et [Clustering: Availability, multi-instance, and disaster recovery](#) pour plus d'informations.

Remarque : Lorsque vous effectuez une mise à niveau vers une version ultérieure de IBM MQ, veillez à effectuer une sauvegarde du fichier `.ini` et des entrées de registre. Les informations du gestionnaire de files d'attente sont stockées dans le fichier `.ini` et peuvent être utilisées pour rétablir une version précédente d'IBM MQ.

Procédure

1. Recherchez les répertoires dans lesquels le gestionnaire de files d'attente place ses données et ses fichiers journaux, à l'aide des informations des fichiers de configuration.
2. Videz les répertoires dans lesquels vous allez placer les données sauvegardées.
3. Copiez les données et les fichiers journaux du gestionnaire de files d'attente sauvegardés aux emplacements appropriés.

Vérifiez que vous disposez d'un fichier de contrôle de journal ainsi que des fichiers journaux.

Pour la journalisation avec réutilisation automatique des journaux, sauvegardez simultanément les répertoires de données et de fichiers journaux du gestionnaire de files d'attente afin de pouvoir restaurer un ensemble cohérent de données et de journaux du gestionnaire de files d'attente.

Pour la journalisation linéaire, sauvegardez en même temps les répertoires des fichiers journaux et des données du gestionnaire de files d'attente. Il est possible de restaurer uniquement les fichiers de données du gestionnaire de files d'attente si une séquence complète correspondante de fichiers journaux est disponible.

4. Mettez à jour les fichiers d'informations de configuration.

Vérifiez que les fichiers de configuration IBM MQ et du gestionnaire de files d'attente sont cohérents afin que IBM MQ puisse rechercher les données restaurées aux emplacements appropriés.

5. Vérifiez la structure de répertoire résultante pour vous assurer que vous disposez de tous les répertoires requis.

Pour plus d'informations sur les répertoires et sous-répertoires IBM MQ, voir [Structure de répertoire sur les systèmes Windows](#) et [Contenu de répertoire sur les systèmes AIX and Linux](#).

Résultats

Si les données ont été sauvegardées et restaurées correctement, le gestionnaire de files d'attente démarre.

Multi

Sauvegarde de la configuration du gestionnaire de files d'attente

La sauvegarde de la configuration du gestionnaire de files d'attente peut vous aider à régénérer un gestionnaire de files d'attente à partir de ses définitions si la configuration du gestionnaire de files d'attente et les données de journal sont perdues en raison d'une panne matérielle et que le gestionnaire de files d'attente ne peut pas redémarrer ou être récupéré à partir du journal.

Pourquoi et quand exécuter cette tâche

ALW

Sous AIX, Linux, and Windows, vous pouvez utiliser la commande **dmpmqcfig** pour vider la configuration d'un gestionnaire de files d'attente IBM MQ.

IBM i

Sous IBM i, vous pouvez utiliser la commande Dump MQ Configuration (**DMPMQMCFG**) pour vider les objets de configuration et les droits d'accès d'un gestionnaire de files d'attente.

Procédure

1. Assurez-vous que le gestionnaire de files d'attente est actif.
2. Selon votre plateforme, utilisez l'une des commandes suivantes pour sauvegarder la configuration du gestionnaire de files d'attente:
 - **ALW** Sous AIX, Linux, and Windows: Exécutez la commande Dump MQ Configuration, **dmpmqcfig**, à l'aide de l'option de formatage par défaut (-f mqsc) MQSC et de tous les attributs (-a), utilisez la redirection de sortie standard pour stocker les définitions dans un fichier. Exemple :

```
dmpmqcfig -m MYQMGR -a > /mq/backups/MYQMGR.mqsc
```

- IBM i Sous IBM i: Exécutez la commande Dump MQ Configuration (**DMPMQMCFG**) à l'aide de l'option de formatage par défaut OUTPUT (*MQSC) et EXPATTR (*ALL), utilisez TOFILE et TOMBR pour stocker les définitions dans un membre de fichier physique. Exemple :

```
DMPMQMCFG MQMNAME(MYQMGR) OUTPUT(*MQSC) EXPATTR(*ALL) TOFILE(QMQMSAMP/QMQSC)
TOMBR(MYQMGRDEF)
```

Tâches associées

«Restauration de la configuration du gestionnaire de files d'», à la page 741

Vous pouvez restaurer la configuration d'un gestionnaire de files d'attente à partir d'une sauvegarde en vous assurant d'abord que le gestionnaire de files d'attente est en cours d'exécution, puis en exécutant la commande appropriée pour votre plateforme.

Référence associée

[dmpmqcfg](#) (vidage de la configuration du gestionnaire de files d'attente)

[Vidage MQ Configuration \(DMPMQMCFG\)](#)

Multi

Restauration de la configuration du gestionnaire de files d'

Vous pouvez restaurer la configuration d'un gestionnaire de files d'attente à partir d'une sauvegarde en vous assurant d'abord que le gestionnaire de files d'attente est en cours d'exécution, puis en exécutant la commande appropriée pour votre plateforme.

Pourquoi et quand exécuter cette tâche

ALW

Sous AIX, Linux, and Windows, vous pouvez utiliser la commande **runmqsc** pour restaurer la configuration d'un gestionnaire de files d'attente IBM MQ .

IBM i

Sous IBM i, vous pouvez utiliser la commande **STRMQMMQSC** pour restaurer les objets de configuration et les droits d'accès d'un gestionnaire de files d'attente.

Procédure

1. Assurez-vous que le gestionnaire de files d'attente est actif.

Notez que si des dommages aux données et aux journaux sont irrémédiables par d'autres moyens, le gestionnaire de files d'attente peut avoir été recréé.

2. Selon votre plateforme, utilisez l'une des commandes suivantes pour restaurer la configuration du gestionnaire de files d'attente:

ALW

- ALW Sous AIX, Linux, and Windows, exécutez **runmqsc** sur le gestionnaire de files d'attente, utilisez la redirection d'entrée standard pour restaurer les définitions à partir d'un fichier script généré par la commande Dump MQ Configuration (**dmpmqcfg**) (voir «Sauvegarde de la configuration du gestionnaire de files d'attente», à la page 740). Exemple :

```
runmqsc MYQMGR < /mq/backups/MYQMGR.mqsc
```

IBM i

- IBM i Sous IBM i: exécutez **STRMQMMQSC** sur le gestionnaire de files d'attente et utilisez les paramètres **SRCMBR** et **SRCFILE** pour restaurer les définitions à partir du membre de fichier physique généré par la commande Dump MQ Configuration (**DMPMQMCFG**) (voir «Sauvegarde de la configuration du gestionnaire de files d'attente», à la page 740). Exemple :

```
STRMQMMQSC MQMNAME(MYQMGR) SRCFILE(QMQMSAMP/QMQSC) SRCMBR(MYQMGR)
```

Tâches associées

«Sauvegarde de la configuration du gestionnaire de files d'attente», à la page 740

La sauvegarde de la configuration du gestionnaire de files d'attente peut vous aider à régénérer un gestionnaire de files d'attente à partir de ses définitions si la configuration du gestionnaire de files d'attente et les données de journal sont perdues en raison d'une panne matérielle et que le gestionnaire de files d'attente ne peut pas redémarrer ou être récupéré à partir du journal.

Référence associée

[dmpmqcfg \(vidage de la configuration du gestionnaire de files d'attente\)](#)

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

[Vidage MQ Configuration \(DMPMQMCFG\)](#)

[Commandes IBM MQ de démarrage \(STRMQMMQSC\)](#)

Utilisation d'un gestionnaire de files d'attente de sauvegarde

Un gestionnaire de files d'attente existant peut disposer d'un gestionnaire de files d'attente de sauvegarde dédié à des fins de reprise après incident.

Pourquoi et quand exécuter cette tâche

Un gestionnaire de files d'attente de sauvegarde est une copie inactive du gestionnaire de files d'attente existant. Si le gestionnaire de files d'attente existant devient irrécupérable en raison d'une défaillance matérielle grave, le gestionnaire de files d'attente de sauvegarde peut être mis en ligne pour remplacer le gestionnaire de files d'attente irrécupérable.

Les fichiers journaux de gestionnaire de files d'attente existants doivent être régulièrement copiés dans le gestionnaire de files d'attente de sauvegarde afin de garantir que le gestionnaire de files d'attente de sauvegarde reste une méthode efficace pour la reprise après incident. The existing queue manager does not need to be stopped for log files to be copied, however you should only copy a log file if the queue manager has finished writing to it; see [«mise à jour d'un gestionnaire de files d'attente de sauvegarde»](#), à la page 743 for information on how to ensure a specific log file is not being written to anymore, so that it can be safely copied.

Remarque : Etant donné que le journal du gestionnaire de files d'attente existant est continuellement mis à jour, il existe toujours une légère différence entre le journal du gestionnaire de files d'attente existant et les données de journal copiées dans le journal du gestionnaire de files d'attente de sauvegarde. Les mises à jour régulières du gestionnaire de files d'attente de sauvegarde réduisent la différence entre les deux journaux.

Si un gestionnaire de files d'attente de sauvegarde doit être mis en ligne, il doit être activé, puis démarré. L'activation d'un gestionnaire de files d'attente de sauvegarde avant son démarrage est une mesure préventive de protection contre le démarrage accidentel d'un gestionnaire de files d'attente de sauvegarde. Une fois qu'un gestionnaire de files d'attente de sauvegarde est activé, il ne peut plus être mis à jour.

Important : Une fois que l'ancien gestionnaire de files d'attente de sauvegarde est devenu le nouveau gestionnaire de files d'attente actif, pour quelque raison que ce soit, il n'existe plus de gestionnaire de files d'attente de sauvegarde. Il s'agit en fait d'une forme de réplication asynchrone, de sorte que le nouveau gestionnaire de files d'attente actif est censé être logiquement un peu en retard sur l'ancien gestionnaire de files d'attente actif. Par conséquent, l'ancien gestionnaire de files d'attente actif n'agit plus comme une sauvegarde du nouveau gestionnaire de files d'attente actif.

Procédure

- Pour plus d'informations sur l'utilisation d'un gestionnaire de files d'attente de sauvegarde, voir les rubriques suivantes:
 - [«Création d'un gestionnaire de files d'attente de sauvegarde»](#), à la page 743
 - [«mise à jour d'un gestionnaire de files d'attente de sauvegarde»](#), à la page 743
 - [«Démarrage d'un gestionnaire de files d'attente de sauvegarde»](#), à la page 744

Concepts associés

«Consignation: S'assurer que les messages ne sont pas perdus», à la page 703

IBM MQ enregistre toutes les modifications importantes apportées aux données persistantes contrôlées par le gestionnaire de files d'attente dans un journal de reprise.

Création d'un gestionnaire de files d'attente de sauvegarde

Vous créez un gestionnaire de files d'attente de sauvegarde en tant que copie inactive du gestionnaire de files d'attente existant.

Pourquoi et quand exécuter cette tâche

Important : Vous ne pouvez utiliser un gestionnaire de files d'attente de sauvegarde que si vous utilisez la journalisation linéaire.

Un gestionnaire de files d'attente de sauvegarde requiert les éléments suivants:

- Pour avoir les mêmes attributs que le gestionnaire de files d'attente existant, par exemple le nom du gestionnaire de files d'attente, le type de consignation et la taille du fichier journal.
- Doit se trouver sur la même plateforme que le gestionnaire de files d'attente existant.
- Doit être à un niveau de code égal ou supérieur à celui du gestionnaire de files d'attente existant.

Procédure

1. Créez un gestionnaire de files d'attente de sauvegarde pour le gestionnaire de files d'attente existant à l'aide de la commande de contrôle **crtmqm**.
2. Effectuez des copies de tous les répertoires de données et de fichiers journaux du gestionnaire de files d'attente existant, y compris de tous les sous-répertoires, comme décrit dans «Sauvegarde des données de gestionnaire de files d'attente», à la page 738.
3. Remplacez les répertoires de données et de fichiers journaux du gestionnaire de files d'attente de sauvegarde, y compris tous les sous-répertoires, par les copies provenant du gestionnaire de files d'attente existant.
4. Exécutez la commande de contrôle **strmqm** sur le gestionnaire de files d'attente de sauvegarde, comme illustré dans l'exemple suivant:

```
strmqm -r BackupQMName
```

Cette commande marque le gestionnaire de files d'attente en tant que gestionnaire de files d'attente de sauvegarde dans IBM MQ et réexécute toutes les extensions de journal copiées pour mettre le gestionnaire de files d'attente de sauvegarde en phase avec le gestionnaire de files d'attente existant.

Référence associée

[crtmqm \(création du gestionnaire de files d'attente\)](#)

[strmqm \(démarrage d'un gestionnaire de files d'attente\)](#)

mise à jour d'un gestionnaire de files d'attente de sauvegarde

Pour garantir qu'un gestionnaire de files d'attente de sauvegarde reste une méthode efficace pour la reprise après incident, il doit être mis à jour régulièrement.

Pourquoi et quand exécuter cette tâche

La mise à jour régulière réduit la différence entre le journal du gestionnaire de files d'attente de sauvegarde et le journal du gestionnaire de files d'attente en cours. Il n'est pas nécessaire d'arrêter le gestionnaire de files d'attente avant de le sauvegarder.



Avertissement : Si vous copiez un ensemble non contigu de journaux dans le répertoire des journaux du gestionnaire de files d'attente de sauvegarde, seuls les journaux jusqu'au point où le premier journal manquant est trouvé sont réexécutés.

Procédure

1. Emettez la commande de script (MQSC) suivante sur le gestionnaire de files d'attente à sauvegarder:

```
RESET QMGR TYPE(ADVANCELOG)
```

Cette opération arrête toute écriture dans le journal en cours, puis avance la consignation du gestionnaire de files d'attente dans le domaine réservé au journal suivant. Cela vous permet de sauvegarder toutes les informations consignées à l'heure en cours.

2. Obtenez le (nouveau) numéro d'extension du journal actif en cours en émettant la commande MQSC (Script) suivante sur le gestionnaire de files d'attente à sauvegarder:

```
DIS QMSTATUS CURRLOG
```

3. Copiez les fichiers d'extension de journal mis à jour du répertoire de journaux du gestionnaire de files d'attente en cours vers le répertoire de journaux du gestionnaire de files d'attente de sauvegarde.
Copiez tous les domaines réservés aux journaux depuis la dernière mise à jour et jusqu'au domaine en cours indiqué dans «2», à la page 744 (non compris). Copiez uniquement les fichiers d'extension de journal, ceux commençant par "S. ..".
4. Exécutez la commande de contrôle **strmqm** sur le gestionnaire de files d'attente de sauvegarde, comme illustré dans l'exemple suivant:

```
strmqm -r BackupQMName
```

Cette opération réexécute toutes les extensions de journal copiées et met le gestionnaire de files d'attente de sauvegarde en phase avec le gestionnaire de files d'attente. Une fois la réexécution terminée, vous recevez un message qui identifie toutes les extensions de journal requises pour la reprise par redémarrage, ainsi que toutes les extensions de journal requises pour la reprise sur support.

Référence associée

[RESET QMGR](#)

[STATUT QMSTATUS D'AFFICHAGE](#)

[strmqm \(démarrage d'un gestionnaire de files d'attente\)](#)

Démarrage d'un gestionnaire de files d'attente de sauvegarde

Vous pouvez remplacer un gestionnaire de files d'attente de sauvegarde par un gestionnaire de files d'attente irrécupérable.

Pourquoi et quand exécuter cette tâche

Lors de la restauration d'une sauvegarde d'un gestionnaire de files d'attente dans un cluster, voir [«Récupération d'un gestionnaire de files d'attente de cluster»](#), à la page 400 et [Clustering: Availability, multi-instance, and disaster recovery](#) pour plus d'informations.

Si un gestionnaire de files d'attente irrécupérable possède un gestionnaire de files d'attente de sauvegarde dédié, vous pouvez l'activer à la place du gestionnaire de files d'attente irrécupérable.

Lorsqu'un gestionnaire de files d'attente irrécupérable est remplacé par un gestionnaire de files d'attente de sauvegarde, certaines données du gestionnaire de files d'attente irrécupérables peuvent être perdues. La quantité de données perdues dépend de la date de la dernière mise à jour du gestionnaire de files d'attente de sauvegarde. Plus la dernière mise à jour est récente, moins la perte de données du gestionnaire de files d'attente est importante.

Remarque : Même si les données du gestionnaire de files d'attente et les fichiers journaux se trouvent dans des répertoires différents, veillez à les sauvegarder et à les restaurer en même temps. Si les données du gestionnaire de files d'attente et les fichiers journaux ont des âges différents, le gestionnaire de files d'attente n'est pas dans un état valide et ne démarrera probablement pas. Même si elle démarre, vos données risquent d'être corrompues.

Procédure

1. Exécutez la commande de contrôle **strmqm** pour activer le gestionnaire de files d'attente de sauvegarde, comme illustré dans l'exemple suivant:

```
strmqm -a BackupQMName
```

Le gestionnaire de files d'attente de sauvegarde est activé. Maintenant qu'il est actif, le gestionnaire de files d'attente de sauvegarde ne peut plus être mis à jour.

2. Exécutez la commande de contrôle **strmqm** pour démarrer le gestionnaire de files d'attente de sauvegarde, comme illustré dans l'exemple suivant:

```
strmqm BackupQMName
```

IBM MQ considère cette opération comme une reprise par redémarrage et utilise le journal du gestionnaire de files d'attente de sauvegarde. Lors de la dernière mise à jour du gestionnaire de files d'attente de sauvegarde, une réexécution a eu lieu, par conséquent, seules les transactions actives du dernier point de contrôle enregistré sont annulées.

3. Redémarrez tous les canaux.
4. Vérifiez la structure de répertoire résultante pour vous assurer que vous disposez de tous les répertoires requis.
Pour plus d'informations sur les répertoires et sous-répertoires IBM MQ, voir [Prise en charge du système de fichiers de planification](#).
5. Vérifiez que vous disposez d'un fichier de contrôle de journal ainsi que des fichiers journaux. Vérifiez également que les fichiers de configuration IBM MQ et du gestionnaire de files d'attente sont cohérents afin que IBM MQ puisse rechercher les données restaurées aux emplacements appropriés.

Résultats

Si les données ont été sauvegardées et restaurées correctement, le gestionnaire de files d'attente démarre à présent.

Tâches associées

«Redémarrage des canaux arrêtés», à la page 256

Lorsqu'un canal passe à l'état STOPPED, vous devez le redémarrer manuellement.

Référence associée

[strmqm \(démarrage d'un gestionnaire de files d'attente\)](#)

Modifications apportées à la reprise sur incident de cluster sur les serveurs sur Multiplatforms

Le gestionnaire de files d'attente réexécute les opérations qui ont provoqué des problèmes, jusqu'à ce que les problèmes soient résolus. Si, au bout de cinq jours, les problèmes ne sont pas résolus, le gestionnaire de files d'attente s'arrête afin d'éviter que la mémoire cache ne devienne encore plus obsolète.

Le gestionnaire de files d'attente réexécute les opérations qui ont provoqué des problèmes, jusqu'à ce que les problèmes soient résolus. Si, au bout de cinq jours, les problèmes ne sont pas résolus, le gestionnaire de files d'attente s'arrête afin d'éviter que la mémoire cache ne devienne encore plus obsolète. Au fur et à mesure que la mémoire cache devient plus obsolète, cela génère un plus grand nombre de problèmes.

Tous les aspects de la gestion de cluster sont gérés pour un gestionnaire de files d'attente par le processus du gestionnaire de référentiel local, `amqrmfa`. Le processus s'exécute sur tous les gestionnaires de files d'attente, même s'il n'y a pas de définitions de cluster.

IBM MQ, plutôt que d'arrêter le gestionnaire de référentiels et de continuer sans lui, le gestionnaire de référentiels réexécute les opérations ayant échoué. Si le gestionnaire de files d'attente détecte un problème avec le gestionnaire de référentiels, il adopte une des deux lignes de conduite.

1. Si l'erreur ne compromet pas l'opération du gestionnaire de files d'attente, ce dernier écrit un message dans le journal des erreurs. Il réexécute les opérations ayant échoué toutes les 10 minutes jusqu'à ce que l'opération réussisse. Par défaut, vous avez cinq jours pour traiter l'erreur. Si au bout de ces cinq jours vous ne l'avez pas fait, le gestionnaire de files d'attente écrit un message dans le journal d'erreur et s'arrête. Vous pouvez reporter l'arrêt de cinq jours.
2. Si l'erreur compromet l'opération du gestionnaire de files d'attente, ce dernier écrit un message dans le journal des erreurs, et s'arrête immédiatement.

Une erreur qui compromet l'opération du gestionnaire de files d'attente est une erreur que le gestionnaire de files d'attente n'a pas été en mesure de diagnostiquer, ou une erreur qui risque d'avoir des conséquences imprévisibles. Très souvent, suite à ce type d'erreur, le gestionnaire de files d'attente écrit un fichier FFST. Il se peut que les erreurs qui compromettent l'opération du gestionnaire de files d'attente soient causés par un bogue dans IBM MQ, ou par un administrateur, ou un programme, en faisant quelque chose d'inattendu, tel que mettre fin à un processus IBM MQ.

Le point du changement dans le comportement de la reprise est de limiter le temps que le gestionnaire de files d'attente continue de s'exécuter avec un nombre croissant de définitions incohérentes de cluster. Au fur et à mesure que les incohérences dans les définitions de cluster augmentent, la possibilité d'un comportement anormal de l'application augmente également.

Le choix par défaut d'arrêter le gestionnaire de files d'attente après cinq jours est un compromis entre limiter le nombre d'incohérences et garder le gestionnaire de files d'attente disponible jusqu'à ce que les problèmes soient détectés et résolus.

Vous pouvez prolonger indéfiniment le temps avant que le gestionnaire de files d'attente ne s'arrête, pendant que vous corrigez le problème ou que vous attendez un arrêt planifié du gestionnaire de files d'attente. Le délai de cinq jours permet au gestionnaire de files d'attente de rester en exécution pendant un long week-end, vous donnant ainsi du temps pour réagir aux éventuels problèmes ou de prolonger le délai avant de redémarrer le gestionnaire de files d'attente.

Actions correctives

Vous disposez de toute une gamme d'actions pour gérer les problèmes de reprise sur incident de cluster. Le premier choix consiste à surveiller et à résoudre le problème et le second à surveiller et à différer la résolution du problème.

1. Analyser le journal des erreurs du gestionnaire de files d'attente à la recherche de messages d'erreur [AMQ9448](#) et [AMQ5008](#), et résoudre le problème.

[AMQ9448](#) indique que le gestionnaire de référentiels a renvoyé une erreur après avoir exécuté une commande. Cette erreur marque le début d'un nouvel essai avec la commande chaque 10 minutes, et éventuellement l'arrêt du gestionnaire de files d'attente après cinq jours, sauf si vous renvoyez l'arrêt.

[AMQ5008](#) indique que le gestionnaire de files d'attente était arrêté car un processus IBM MQ est manquant. [AMQ5008](#) résulte de l'arrêt du gestionnaire de référentiels après cinq jours. Si le gestionnaire de files d'attente s'arrête, le gestionnaire de files d'attente s'arrête.

2. Analyser le journal des erreurs du gestionnaire de files d'attente à la recherche du message d'erreur [AMQ9448](#), et reporter la résolution du problème.

Si vous désactivez la réception des messages à partir de `SYSTEM.CLUSTER.COMMAND.QUEUE`, le gestionnaire de référentiels cesse de tenter d'exécuter des commandes, et continue indéfiniment sans exécuter aucun travail. Toutefois, les descripteurs maintenus sur les files d'attente par le gestionnaire de référentiels sont libérés. Etant donné que le gestionnaire de référentiels ne s'arrête pas, le gestionnaire de files d'attente ne s'arrête pas après cinq jours.

Exécutez une commande MQSC pour désactiver la réception de messages à partir de `SYSTEM.CLUSTER.COMMAND.QUEUE` :

```
ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(DISABLED)
```

Pour reprendre la réception des messages depuis `SYSTEM.CLUSTER.COMMAND.QUEUE`, exécutez une commande MQSC :

```
ALTER QLOCAL(SYSTEM.CLUSTER.COMMAND.QUEUE) GET(ENABLED)
```

Considération spéciale

L'arrêt de `amqrrmfa` dans IBM MQ amène l'arrêt du gestionnaire de files d'attente, car c'est considéré comme un échec du gestionnaire de files d'attente. Vous ne devez pas arrêter le processus `amqrrmfa` sauf si vous définissez le paramètre d'optimisation du gestionnaire de files d'attente, `TolerateRepositoryFailure`.

Exemple

```
TuningParameters:  
TolerateRepositoryFailure=TRUE
```

Figure 86. Définissez `TolerateRepositoryFailure` sur `TRUE` dans `qm.ini`

Concepts associés

«Fichiers de configuration du gestionnaire de files d'attente, `qm.ini`», à la page 111

Un fichier de configuration de gestionnaire de files d'attente, `qm.ini`, contient des informations relatives à un gestionnaire de files d'attente spécifique. Les attributs que vous pouvez utiliser pour modifier la configuration d'un gestionnaire de files d'attente individuel remplacent tous les paramètres d'IBM MQ.

Configuration des ressources JMS et Jakarta Messaging

L'une des façons dont une application JMS ou Jakarta Messaging peut créer et configurer les ressources dont elle a besoin pour se connecter à IBM MQ et accéder aux destinations pour l'envoi ou la réception de messages consiste à utiliser l'interface JNDI (Java Naming and Directory Interface) pour extraire des objets gérés à partir d'un emplacement du service d'annuaire appelé espace de nom JNDI. Pour qu'une application JMS puisse extraire des objets gérés d'un espace de nom JNDI, vous devez d'abord créer et configurer les objets gérés.

Pourquoi et quand exécuter cette tâche

JM 3.0 Depuis IBM MQ 9.3.0, Jakarta Messaging 3.0 est pris en charge pour le développement de nouvelles applications. IBM MQ 9.3.0 et les versions ultérieures continuent de prendre en charge JMS 2.0 pour les applications existantes. L'utilisation de l'API Jakarta Messaging 3.0 et de l'API JMS 2.0 dans la même application n'est pas prise en charge. Pour plus d'informations, voir [Utilisation des classes IBM MQ pour JMS/Jakarta Messaging](#).

Vous pouvez créer et configurer des objets gérés dans IBM MQ à l'aide de l'un des outils suivants:

Outils d'administration IBM MQ JMS et Jakarta Messaging

L'outil d'administration IBM MQ JMS, **JMSAdmin**, et l'outil d'administration Jakarta Messaging, **JMS30Admin**, sont des outils de ligne de commande que vous pouvez utiliser pour créer et configurer des objets IBM MQ JMS et Jakarta Messaging stockés dans LDAP, dans un système de fichiers local ou dans d'autres emplacements. Les outils d'administration JMS et Jakarta Messaging utilisent une syntaxe similaire à `runmqsc` et prennent également en charge le scriptage.

Les outils d'administration utilisent un fichier de configuration pour définir les valeurs de certaines propriétés. Un exemple de fichier de configuration est fourni, que vous pouvez éditer pour l'adapter à votre système avant de commencer à l'aide de l'outil de configuration des ressources JMS. Pour plus d'informations sur le fichier de configuration, voir «[Configuration des outils JMSAdmin et JMS30Admin](#)», à la page 755.

JMS 2.0 IBM MQ Explorer

Pour JMS 2.0, vous pouvez utiliser IBM MQ Explorer pour créer et administrer des définitions d'objet JMS 2.0 qui sont stockées dans LDAP, dans un système de fichiers local ou dans d'autres emplacements.

JM 3.0 Pour Jakarta Messaging 3.0, vous ne pouvez pas administrer JNDI à l'aide de IBM MQ Explorer. L'administration JNDI est prise en charge par la variante Jakarta Messaging 3.0 de **JMSAdmin**, qui est **JMS30Admin**.

Les applications IBM MQ JMS déployées dans WebSphere Application Server doivent accéder aux objets JMS à partir du référentiel JNDI du serveur d'applications. Par conséquent, si vous utilisez la messagerie JMS entre WebSphere Application Server et IBM MQ, vous devez créer des objets dans WebSphere Application Server qui correspondent aux objets que vous créez dans IBM MQ.

JM 3.0 Bien que IBM MQ 9.3 et les versions ultérieures prennent en charge [Jakarta Messaging 3.0](#), WebSphere Application Server ne dispose pas actuellement d'une prise en charge équivalente. Par conséquent, dans WebSphere Application Server, vous configurez les ressources Java Message Service 2.0 .

IBM MQ Explorer et l'outil d'administration IBM MQ JMS ne peuvent pas être utilisés pour administrer les objets IBM MQ JMS stockés dans WebSphere Application Server. A la place, vous pouvez créer et configurer des objets gérés dans WebSphere Application Server à l'aide de l'un des outils suivants:

WebSphere Application Server console d'administration

La console d'administration WebSphere Application Server est un outil Web que vous pouvez utiliser pour gérer les objets IBM MQ JMS dans WebSphere Application Server.

Client de scriptage wsadmin WebSphere Application Server

Le client de scriptage wsadmin WebSphere Application Server fournit des commandes spécialisées pour administrer des objets IBM MQ JMS dans WebSphere Application Server.

Si vous souhaitez utiliser une application JMS pour accéder aux ressources d'un gestionnaire de files d'attente IBM MQ depuis WebSphere Application Server, vous devez utiliser le fournisseur de messagerie IBM MQ dans WebSphere Application Server, qui contient une version de IBM MQ classes for JMS. L'adaptateur de ressources IBM MQ fourni avec WebSphere Application Server est utilisé par toutes les applications qui exécutent la messagerie JMS avec le fournisseur de messagerie IBM MQ . L'adaptateur de ressources IBM MQ est généralement mis à jour automatiquement lorsque vous appliquez des groupes de correctifs WebSphere Application Server , mais si vous avez précédemment mis à jour manuellement l'adaptateur de ressources, vous devez mettre à jour manuellement votre configuration pour vous assurer que la maintenance est correctement appliquée.

Concepts associés

[Création et configuration de fabriques de connexions et de destinations dans une application IBM MQ classes for JMS](#)

Référence associée

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

Configuration de fabriques de connexions et de destinations dans un espace de nom JNDI

Les applications JMS et Jakarta Messaging accèdent aux objets gérés dans le service de nommage et d'annuaire via l'interface JNDI (Java Naming and Directory Interface). Les objets administrés JMS ou Jakarta Messaging sont stockés dans un emplacement du service d'annuaire appelé espace de nom JNDI. Une application JMS ou Jakarta Messaging peut rechercher les objets gérés pour se connecter à IBM MQ et accéder à des destinations pour l'envoi ou la réception de messages.

Pourquoi et quand exécuter cette tâche

Les applications JMS ou Jakarta Messaging recherchent les noms des objets JMS ou Jakarta Messaging dans le service d'annuaire en utilisant les contextes suivants:

Contexte initial

Le contexte initial définit la racine de l'espace de nom JNDI. Pour chaque emplacement du service d'annuaire, vous devez spécifier un contexte initial pour fournir un point de départ à partir duquel

une application JMS ou Jakarta Messaging peut résoudre les noms des objets gérés dans cet emplacement du service d'annuaire.

Sous-contextes

Un contexte peut comporter un ou plusieurs sous-contextes. Un sous-contexte est une subdivision d'un espace de nom JNDI et peut contenir des objets gérés tels que des fabriques de connexions et des destinations, ainsi que d'autres sous-contextes. Un sous-contexte n'est pas un objet à part entière ; il s'agit simplement d'une extension de la convention de dénomination applicable aux objets du sous-contexte.

Pour qu'une application IBM MQ classes for JMS ou IBM MQ classes for Jakarta Messaging puisse extraire des objets gérés d'un espace de nom JNDI, vous devez d'abord créer les objets gérés. Vous pouvez créer et configurer les types d'objet JMS ou Jakarta Messaging suivants:

Fabrique de connexions

Un objet de fabrique de connexions JMS ou Jakarta Messaging définit un ensemble de propriétés de configuration standard pour les connexions. Une application JMS ou Jakarta Messaging utilise une fabrique de connexions pour créer une connexion à IBM MQ. Vous pouvez créer une fabrique de connexions spécifique à l'un des deux domaines de messagerie, le domaine de messagerie point-à-point et le domaine de messagerie de publication / abonnement.

Sinon, à partir de JMS 1.1, vous pouvez créer des fabriques de connexions indépendantes du domaine qui peuvent être utilisées pour la messagerie point-à-point et la messagerie de publication / abonnement. Pour plus d'informations, voir [Le modèle de messagerie JMS et Jakarta](#).

Destination

Une destination JMS ou Jakarta Messaging est un objet qui représente la cible des messages produits par le client et la source des messages consommés par une application JMS . L'application JMS ou Jakarta Messaging peut utiliser un objet de destination unique pour insérer des messages et en extraire des messages, ou l'application peut utiliser des objets de destination distincts. Il existe deux types d'objet de destination:

- Destination de file d'attente JMS ou Jakarta Messaging utilisée dans la messagerie point-à-point
- Destination de rubrique JMS ou Jakarta Messaging utilisée dans la messagerie de publication / abonnement

JMS 2.0 Pour JMS 2.0, vous pouvez créer des contextes et des objets gérés à l'aide de IBM MQ Explorer ou de l'outil d'administration IBM MQ **JMSAdmin**.

Remarque : **JM 3.0** Pour Jakarta Messaging 3.0, vous ne pouvez pas administrer JNDI à l'aide de IBM MQ Explorer. L'administration JNDI est prise en charge par la variante Jakarta Messaging 3.0 de **JMSAdmin**, qui est **JMS30Admin**.

Le diagramme suivant illustre un exemple d'objets JMS ou Jakarta Messaging créés dans un espace de nom JNDI IBM MQ .

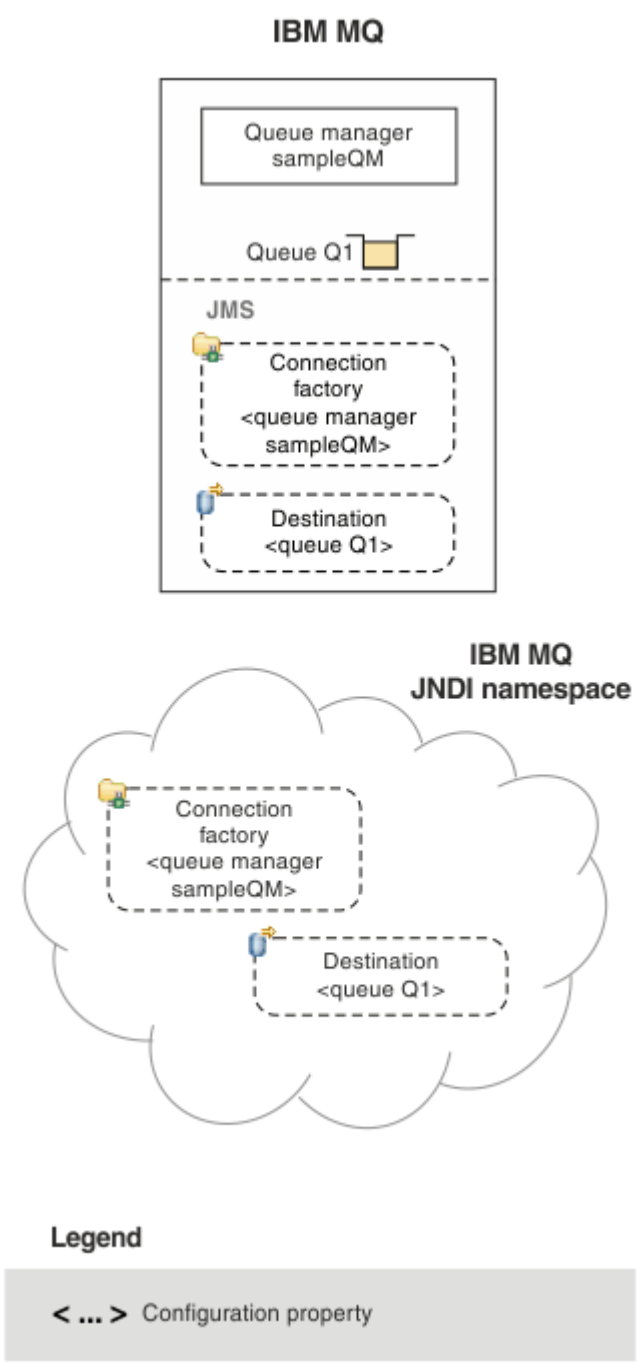


Figure 87. Objets JMS ou Jakarta Messaging créés dans IBM MQ

Si vous utilisez la messagerie JMS entre WebSphere Application Server et IBM MQ, vous devez créer des objets correspondants dans WebSphere Application Server pour communiquer avec IBM MQ. Lorsque vous créez l'un de ces objets dans WebSphere Application Server, il est stocké dans l'espace de nom JNDI WebSphere Application Server, comme illustré dans le diagramme suivant.

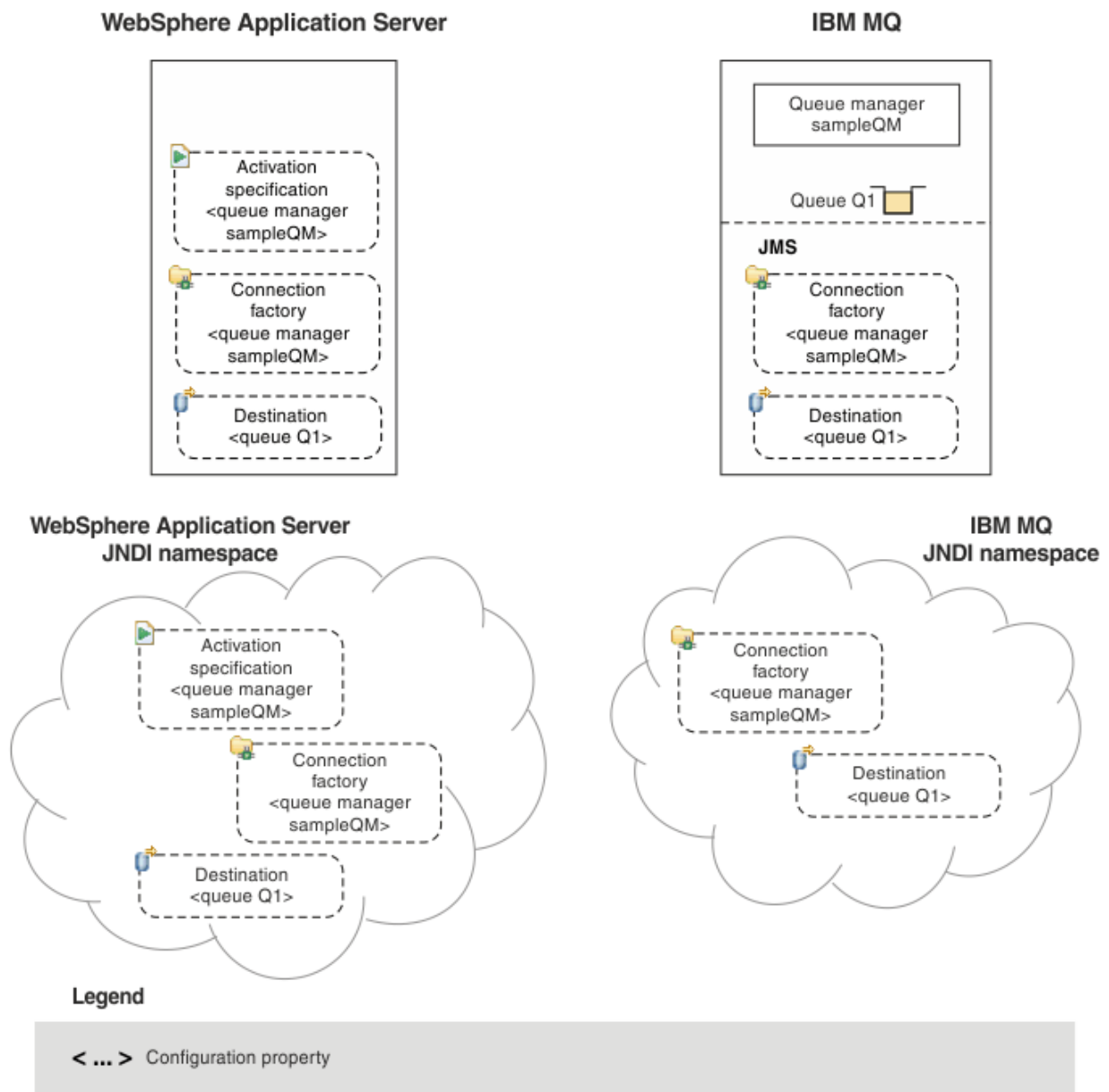


Figure 88. Objets créés dans WebSphere Application Server et objets correspondants dans IBM MQ

JM 3.0 Bien que IBM MQ 9.3 et les versions ultérieures prennent en charge Jakarta Messaging 3.0, WebSphere Application Server ne dispose pas actuellement d'une prise en charge équivalente. Par conséquent, dans WebSphere Application Server, vous configurez les ressources Java Message Service 2.0.

Si votre application utilise un bean géré par message (MDB), la fabrique de connexions est utilisée pour les messages sortants uniquement et les messages entrants sont reçus par une spécification d'activation. Les spécifications d'activation font partie de la norme Java EE Connector Architecture 1.5 (JCA 1.5). JCA 1.5 fournit un moyen standard d'intégrer des fournisseurs JMS, tels que IBM MQ, à des serveurs d'applications Java EE, tels que WebSphere Application Server. Une spécification d'activation JMS peut être associée à un ou plusieurs beans gérés par message (MDB) et fournit la configuration nécessaire pour que ces MDB soient à l'écoute des messages arrivant à une destination.

Vous pouvez utiliser la console d'administration WebSphere Application Server ou les commandes de scriptage wsadmin pour créer et configurer les ressources JMS dont vous avez besoin.

Procédure

- **JMS 2.0**
Pour configurer des objets JMS pour IBM MQ à l'aide de IBM MQ Explorer, voir [«Configuration d'objets JMS 2.0 à l'aide de IBM MQ Explorer»](#), à la page 752.
- **JMS 2.0**
Pour configurer des objets JMS pour IBM MQ à l'aide de l'outil d'administration IBM MQ JMS, **JMSAdmin**, voir [«Configuration des objets JMS et Jakarta Messaging à l'aide des outils d'administration»](#), à la page 753.
- **JM 3.0**
Pour configurer des objets Jakarta Messaging pour IBM MQ à l'aide de l'outil d'administration IBM MQ Jakarta Messaging, **JMS30Admin**, voir [«Configuration des objets JMS et Jakarta Messaging à l'aide des outils d'administration»](#), à la page 753.
- **JMS 2.0**
Pour configurer des objets JMS pour WebSphere Application Server, voir [«Configuration des ressources JMS 2.0 dans WebSphere Application Server»](#), à la page 764.

Résultats

Une application IBM MQ classes for JMS ou IBM MQ classes for Jakarta Messaging peut extraire les objets gérés de l'espace de nom JNDI et, si nécessaire, définir ou modifier une ou plusieurs de ses propriétés à l'aide des extensions IBM JMS ou IBM MQ JMS .

Tâches associées

Utilisation de JNDI pour extraire des objets gérés dans une application JMS

[Création et configuration de fabriques de connexions et de destinations dans une application IBM MQ classes for JMS](#)

JMS 2.0 Configuration d'objets JMS 2.0 à l'aide de IBM MQ Explorer

Utilisez l'interface graphique IBM MQ Explorer pour créer des objets JMS à partir d'objets IBM MQ et des objets IBM MQ à partir d'objets JMS , ainsi que pour administrer et surveiller d'autres objets IBM MQ .

Pourquoi et quand exécuter cette tâche

JMS 2.0 IBM MQ Explorer est l'interface utilisateur graphique qui vous permet d'administrer et de surveiller les objets IBM MQ, qu'ils soient hébergés en local sur votre ordinateur ou sur un système distant. IBM MQ Explorer s'exécute sous Windows et Linux for x86-64. Il peut se connecter à distance aux gestionnaires de files d'attente qui s'exécutent sur n'importe quelle plateforme prise en charge, y compris z/OS, ce qui permet d'afficher, d'explorer et de modifier l'intégralité de votre réseau principal de messagerie à partir de la console.

Remarque : **JM 3.0** Pour Jakarta Messaging 3.0, vous ne pouvez pas administrer JNDI à l'aide de IBM MQ Explorer. L'administration JNDI est prise en charge par la variante Jakarta Messaging 3.0 de **JMSAdmin**, qui est **JMS30Admin**.

Dans IBM MQ Explorer, toutes les fabriques de connexions sont stockées dans les dossiers des fabriques de connexions dans le contexte et les sous-contextes appropriés.

Vous pouvez effectuer les types de tâche suivants avec IBM MQ Explorer, soit en contexte à partir d'un objet existant dans IBM MQ Explorer, soit à partir d'un assistant de création d'objet:

- Créez une fabrique de connexions JMS à partir de l'un des objets IBM MQ suivants:
 - Un gestionnaire de files d'attente IBM MQ , que ce soit sur votre ordinateur local ou sur un système distant.
 - Un canal IBM MQ .

- Un programme d'écoute IBM MQ .
- Ajoutez un gestionnaire de files d'attente IBM MQ à IBM MQ Explorer à l'aide d'une fabrique de connexions JMS .
- Créez une file d'attente JMS à partir d'une file d'attente IBM MQ .
- Créez une file d'attente IBM MQ à partir d'une file d'attente JMS .
- Créez une rubrique JMS à partir d'une rubrique IBM MQ , qui peut être un objet IBM MQ ou une rubrique dynamique.
- Créez une rubrique IBM MQ à partir d'une rubrique JMS .

Procédure

- Démarrez IBM MQ Explorer, s'il n'est pas déjà en cours d'exécution.
Si IBM MQ Explorer est déjà en cours d'exécution et affiche la page d'accueil, fermez la page d'accueil pour commencer à gérer les objets IBM MQ.
- Si ce n'est pas déjà fait, créez un contexte initial définissant la racine de l'espace de nom JNDI dans lequel les objets JMS sont stockés dans le service d'annuaire.
Lorsque vous avez ajouté le contexte initial à IBM MQ Explorer, vous pouvez créer des objets fabrique de connexions, des objets destination ainsi que des sous-contextes dans l'espace de nom JNDI.
Le contexte initial s'affiche dans la vue Navigator du dossier Objets gérés par JMS . Notez que même si le contenu complet de l'espace de nom JNDI est affiché, dans IBM MQ Explorer , vous ne pouvez éditer que les objets IBM MQ classes for JMS qui y sont stockés. Pour plus d'informations, voir [Ajout d'un contexte initial](#).
- Créez et configurez les sous-contextes et les objets administrés JMS dont vous avez besoin.
Pour plus d'informations, voir [Création et configuration d'objets gérés JMS](#).
- Configurez IBM MQ.
Pour plus d'informations, voir [Configuration de IBM MQ à l'aide de IBM MQ Explorer](#) .

Concepts associés

[Présentation de IBM MQ Explorer](#)

[Création et configuration de fabriques de connexions et de destinations dans une application IBM MQ classes for JMS](#)

Configuration des objets JMS et Jakarta Messaging à l'aide des outils d'administration

IBM MQ fournit des outils d'administration que vous pouvez utiliser pour définir les propriétés de huit types d'objet IBM MQ classes for JMS ou IBM MQ classes for Jakarta Messaging et pour les stocker dans un espace de nom JNDI. Les applications peuvent ensuite utiliser JNDI pour extraire ces objets gérés de l'espace de nom.

Pourquoi et quand exécuter cette tâche

JMS 2.0 Pour [JMS 2.0](#), l'administration JNDI est prise en charge par l'outil **JMSAdmin** .

JM 3.0 Pour [Jakarta Messaging 3.0](#), l'administration JNDI est prise en charge par la variante Jakarta Messaging 3.0 de **JMSAdmin**, qui est **JMS30Admin**.

Le tableau suivant présente les huit types d'objets gérés que vous pouvez créer, configurer et manipuler à l'aide d'instructions. La colonne Mot clé affiche les chaînes que vous pouvez remplacer par *TYPE* dans les commandes présentées dans [Tableau 37, à la page 754](#).

Tableau 37. Les types d'objet JMS et Jakarta Messaging qui sont gérés par l'outil d'administration

Type d'objet	Mot clé	Description
MQConnectionFactory	CF	Implémentation IBM MQ de l'interface ConnectionFactory de JMS . Il s'agit d'un objet de fabrique permettant de créer des connexions dans les domaines point à point et de publication / abonnement.
Fabrique MQQueueConnection	QCF	Implémentation IBM MQ de l'interface JMS QueueConnectionFactory. Il s'agit d'un objet de fabrique permettant de créer des connexions dans le domaine point-à-point.
Fabrique MQTopicConnection	TCF	Implémentation IBM MQ de l'interface de fabrique JMS TopicConnection. Il s'agit d'un objet de fabrique permettant de créer des connexions dans le domaine de publication / abonnement.
MQQUEUE	Q	Implémentation IBM MQ de l'interface de file d'attente JMS . Il s'agit d'une destination pour les messages du domaine point à point.
Sujet MQ	T	Implémentation IBM MQ de l'interface de rubrique JMS . Il s'agit d'une destination pour les messages du domaine de publication / abonnement.
MQXAConnectionFactory «1», à la page 754	XACF	Implémentation IBM MQ de l'interface JMS XAConnectionFactory . Il s'agit d'un objet de fabrique permettant de créer des connexions dans les domaines point à point et de publication / abonnement, et dans lequel les connexions utilisent les versions XA des classes JMS .
MQXAQueueConnectionMQXAQueueConnec tion «1», à la page 754	XAQCF	Implémentation IBM MQ de l'interface de fabrique JMS XAQueueConnection. Il s'agit d'un objet de fabrique permettant de créer des connexions dans le domaine point à point qui utilisent les versions XA des classes JMS .
MQXATopicConnectionFabrique «1», à la page 754	Fonction XATCF	Implémentation IBM MQ de l'interface JMS XATopicConnectionFactory. Il s'agit d'un objet de fabrique permettant de créer des connexions dans le domaine de publication / abonnement qui utilisent les versions XA des classes JMS .

Remarque :

1. Ces classes sont fournies pour être utilisées par les fournisseurs de serveurs d'applications. Il est peu probable qu'ils soient directement utiles aux programmeurs d'application.

Pour plus d'informations sur la configuration de ces objets, voir [«Configuration des objets JMS»](#), à la page [763](#).

Les types de propriété et les valeurs dont vous avez besoin pour utiliser cet outil sont répertoriés dans la section [Propriétés des objets IBM MQ classes for JMS](#).

Vous pouvez également utiliser l'outil pour manipuler les sous-contextes d'espace de nom de répertoire au sein de l'interface JNDI, comme décrit dans [«Configuration de sous-contextes»](#), à la page 759.

JMS 2.0 Pour JMS 2.0 et les versions antérieures, vous pouvez également créer et configurer des objets gérés IBM MQ classes for JMS avec IBM MQ Explorer.

JM 3.0 Pour Jakarta Messaging 3.0, vous ne pouvez pas administrer JNDI à l'aide de IBM MQ Explorer. L'administration JNDI est prise en charge par la variante Jakarta Messaging 3.0 de **JMSAdmin**, qui est **JMS30Admin**.

Concepts associés

[Création et configuration de fabriques de connexions et de destinations dans une application IBM MQ classes for JMS](#)

[Utilisation de JNDI pour extraire des objets gérés dans une application JMS](#)

Configuration des outils JMSAdmin et JMS30Admin

Les outils d'administration IBM MQ JMS et Jakarta Messaging utilisent un fichier de configuration pour définir les valeurs de certaines propriétés. Dans chaque cas, un exemple de fichier de configuration est fourni et vous pouvez l'éditer pour l'adapter à votre système.

Pourquoi et quand exécuter cette tâche

JM 3.0 IBM MQ 9.3.0 a introduit la prise en charge de [Jakarta Messaging 3.0](#). JMS 2.0 est toujours entièrement pris en charge.

Le fichier de configuration est un fichier en texte clair qui se compose d'un ensemble de paires clé-valeur, séparées par le signe égal (=). Vous configurez l'outil d'administration en définissant des valeurs pour les trois propriétés définies dans le fichier de configuration. L'exemple suivant illustre ces trois propriétés:

```
#Set the service provider
INITIAL_CONTEXT_FACTORY=com.sun.jndi.ldap.LdapCtxFactory
#Set the initial context
PROVIDER_URL=ldap://polaris/o=ibm_us,c=us
#Set the authentication type
SECURITY_AUTHENTICATION=none
```

Dans cet exemple, un signe dièse (#) dans la première colonne de la ligne indique un commentaire ou une ligne qui n'est pas utilisée.

Un exemple de fichier de configuration, utilisé comme fichier de configuration par défaut, est fourni avec IBM MQ. L'exemple de fichier est appelé `JMSAdmin.config` (pour JMS 2.0) ou `JMS30Admin.config` (pour Jakarta Messaging 3.0). Ce fichier se trouve dans le répertoire `MQ_JAVA_INSTALL_PATH/bin`. Vous pouvez soit éditer l'exemple de fichier pour définir les paramètres nécessaires à votre système, soit créer votre propre fichier de configuration.

Lorsque vous démarrez l'outil d'administration, vous pouvez spécifier le fichier de configuration à utiliser à l'aide du paramètre de ligne de commande `-cfg`, comme décrit dans [«Démarrage des outils JMSAdmin et JMS30Admin»](#), à la page 757. Si vous n'indiquez pas de nom de fichier de configuration lorsque vous appelez l'outil, celui-ci tente de charger le fichier de configuration par défaut (`JMSAdmin.config` ou `JMS30Admin.config`). Il recherche d'abord ce fichier dans le répertoire de travail, puis dans le répertoire `MQ_JAVA_INSTALL_PATH/bin`, où `MQ_JAVA_INSTALL_PATH` est le chemin d'accès à votre installation IBM MQ classes for JMS ou IBM MQ classes for Jakarta Messaging.

Les noms des objets JMS ou Jakarta Messaging stockés dans un environnement LDAP doivent être conformes aux conventions de dénomination LDAP. L'une de ces conventions est que les noms d'objet et de contexte doivent inclure un préfixe, tel que `cn=` (nom usuel) ou `ou=` (unité organisationnelle). L'outil d'administration simplifie l'utilisation des fournisseurs de services LDAP en vous permettant de


faire référence à des noms d'objet et de contexte sans préfixe. Si vous n'indiquez pas de préfixe, l'outil ajoute automatiquement un préfixe par défaut au nom que vous indiquez. Pour LDAP, il s'agit de `cn=`. Si nécessaire, vous pouvez modifier le préfixe par défaut en définissant la propriété **NAME_PREFIX** dans le fichier de configuration.

Remarque : Vous devrez peut-être configurer votre serveur LDAP pour stocker des objets Java . Pour plus d'informations, voir la documentation de votre serveur LDAP.

Procédure

1. Définissez le fournisseur de services utilisé par l'outil en configurant la propriété **INITIAL_CONTEXT_FACTORY** .

Les valeurs prises en charge pour cette propriété sont les suivantes:

- `com.sun.jndi.ldap.LdapCtxFactory` (pour LDAP)
- `com.sun.jndi.fscontext.RefFSContextFactory` (pour le contexte du système de fichiers)
-  `com.ibm.jndi.LDAPCtxFactory` est pris en charge sous z/OS uniquement et permet d'accéder à un serveur LDAP. Toutefois, cette classe est incompatible avec `com.sun.jndi.ldap.LdapCtxFactory`, dans la mesure où les objets créés à l'aide d'une fabrique `InitialContext` ne peuvent pas être lus ou modifiés à l'aide de l'autre.

Vous pouvez également utiliser l'outil d'administration pour vous connecter à d'autres contextes JNDI à l'aide de trois paramètres définis dans le fichier de configuration `JMSAdmin` ou `JMS30Admin` . Pour utiliser une autre fabrique `InitialContext`:

- a) Définissez la propriété **INITIAL_CONTEXT_FACTORY** sur le nom de classe requis.
- b) Définissez le comportement de la fabrique `InitialContext` à l'aide des propriétés **USE_INITIAL_DIR_CONTEXT**, **NAME_PREFIX** et **NAME_READABILITY_MARKER** .

Les paramètres de ces propriétés sont décrits dans les exemples de commentaires de fichier de configuration.

Vous n'avez pas besoin de définir les propriétés **USE_INITIAL_DIR_CONTEXT**, **NAME_PREFIX** et **NAME_READABILITY_MARKER** si vous utilisez l'une des valeurs **INITIAL_CONTEXT_FACTORY** prises en charge. Toutefois, vous pouvez attribuer des valeurs à ces propriétés si vous souhaitez remplacer les valeurs par défaut du système. Par exemple, si vos objets sont stockés dans un environnement LDAP, vous pouvez modifier le préfixe par défaut que l'outil ajoute aux noms d'objet et de contexte en définissant la propriété **NAME_PREFIX** sur le préfixe requis.

Si vous omettez une ou plusieurs des trois propriétés de la fabrique `InitialContext`, l'outil d'administration fournit des valeurs par défaut appropriées en fonction des valeurs des autres propriétés.

2. Définissez l'URL du contexte initial de la session en configurant la propriété **PROVIDER_URL** .

Cette URL est la racine de toutes les opérations JNDI effectuées par l'outil. Deux formes de cette propriété sont prises en charge:

- `ldap://hostname/contextname`
- `file: [unité:] /pathname`

Le format de l'URL LDAP peut varier en fonction de votre fournisseur LDAP. Pour plus d'informations, consultez la documentation LDAP.

3. Définissez si JNDI transmet les données d'identification de sécurité à votre fournisseur de services en configurant la propriété **SECURITY_AUTHENTICATION** .

Cette propriété est utilisée uniquement lorsqu'un fournisseur de services LDAP est utilisé et peut prendre l'une des trois valeurs suivantes:

none (authentication anonyme)

Si vous définissez ce paramètre sur `none`, JNDI ne transmet aucune donnée d'identification de sécurité au fournisseur de services et l' *authentication anonyme* est effectuée.

simple (authentification simple)

Si vous définissez le paramètre sur `simple`, les données d'identification de sécurité sont transmises via JNDI au fournisseur de services sous-jacent. Ces données d'identification de sécurité se présentent sous la forme d'un nom distinctif d'utilisateur (nom distinctif d'utilisateur) et d'un mot de passe.

CRAM-MD5 (mécanisme d'authentification CRAM-MD5)

Si vous définissez le paramètre sur `CRAM-MD5`, les données d'identification de sécurité sont transmises via JNDI au fournisseur de services sous-jacent. Ces données d'identification de sécurité se présentent sous la forme d'un nom distinctif d'utilisateur (nom distinctif d'utilisateur) et d'un mot de passe.

Si vous ne fournissez pas de valeur valide pour la propriété **SECURITY_AUTHENTICATION**, la valeur par défaut de la propriété est `none`.

Si des données d'identification de sécurité sont requises, vous êtes invité à les indiquer lors de l'initialisation de l'outil. Vous pouvez éviter cela en définissant les propriétés **PROVIDER_USERDN** et **PROVIDER_PASSWORD** dans le fichier de configuration `JMSAdmin`.

Remarque : Si vous n'utilisez pas ces propriétés, le texte saisi, *incluant le mot de passe*, est renvoyé à l'écran. Cela peut avoir des implications sur la sécurité.

L'outil ne s'authentifie pas lui-même ; la tâche d'authentification est déléguée au serveur LDAP. L'administrateur du serveur LDAP doit configurer et gérer les droits d'accès à différentes parties de l'annuaire. Pour plus d'informations, consultez la documentation LDAP. Si l'authentification échoue, l'outil affiche un message d'erreur approprié et s'arrête.

Des informations plus détaillées sur la sécurité et JNDI sont disponibles sur le site Web Oracle Java ([Oracle Technology Network for Java Developers](#)).

Démarrage des outils JMSAdmin et JMS30Admin

Les outils d'administration IBM MQ JMS et Jakarta Messaging disposent d'une interface de ligne de commande que vous pouvez utiliser de manière interactive ou pour démarrer un traitement par lots.

Pourquoi et quand exécuter cette tâche

Le mode interactif fournit une invite de commande dans laquelle vous pouvez entrer des commandes d'administration. En mode de traitement par lots, la commande de démarrage de l'outil inclut le nom d'un fichier contenant un script de commande d'administration.

Procédure

Mode interactif

- Pour démarrer l'outil en mode interactif, entrez la commande suivante:

```
> JMS 2.0
```

```
JMSAdmin [-t] [-v] [-cfg config_filename]
```

```
> JM 3.0
```

```
JMS30Admin [-t] [-v] [-cfg config_filename]
```

où :

-t

Active la fonction de trace (par défaut, la fonction de trace est désactivée).

Le fichier de trace est généré dans `"%MQ_JAVA_DATA_PATH%\errors` (Windows) ou `/var/mqm/trace` (AIX and Linux). Le nom du fichier de trace est au format suivant:

```
mqjms_PID.trc
```

où *PID* est l'ID de processus de la machine virtuelle Java.

-v

Produit une sortie prolixe (par défaut, il s'agit d'une sortie de type terse).

-cfg nom_fichier_config

Nomme un autre fichier de configuration. Si ce paramètre est omis, le fichier de configuration par défaut, `JMSAdmin.config` (pour JMS 2.0) ou `JMS30Admin.config` (pour Jakarta Messaging 3.0), est utilisé. Pour plus d'informations sur le fichier de configuration, voir «[Configuration des outils JMSAdmin et JMS30Admin](#)», à la page 755.

Une invite de commande s'affiche, indiquant que l'outil est prêt à accepter les commandes d'administration. Cette invite apparaît initialement comme suit:

```
InitCtx>
```

indiquant que le contexte en cours (c'est-à-dire le contexte JNDI auquel toutes les opérations de renommage et d'annuaire font actuellement référence) est le contexte initial défini dans le paramètre de configuration **PROVIDER_URL**. Pour plus d'informations sur ce paramètre, voir «[Configuration des outils JMSAdmin et JMS30Admin](#)», à la page 755.

Au fur et à mesure que vous parcourez l'espace de nom du répertoire, l'invite change pour refléter cela, de sorte que l'invite affiche toujours le contexte en cours.

Mode de traitement par lots

- Pour démarrer l'outil en mode de traitement par lots, entrez la commande suivante:

```
> JMS 2.0
```

```
JMSAdmin test.scp
```

```
> JM 3.0
```

```
JMS30Admin test.scp
```

où `test.scp` est un fichier script qui contient des commandes d'administration. Pour plus d'informations, voir «[Utilisation des commandes d'administration avec JMSAdmin et JMS30Admin](#)», à la page 758. La dernière commande du fichier doit être la commande END.

Utilisation des commandes d'administration avec JMSAdmin et JMS30Admin

Les outils d'administration IBM MQ JMS et Jakarta Messaging acceptent des commandes composées d'une instruction d'administration et de ses paramètres appropriés.

Pourquoi et quand exécuter cette tâche

Le tableau suivant répertorie les instructions d'administration que vous pouvez utiliser lorsque vous entrez des commandes avec les outils d'administration.

Verbe	Forme abrégée	Description
ALTER	ALT	Modifier au moins une des propriétés d'un objet géré
DEFINIR	infrastructure d'évaluation de déploiement	Créer et stocker un objet géré ou créer un sous-contexte

Tableau 38. Instructions d'administration (suite)

Verbe	Forme abrégée	Description
affichage	Data Interchange Services	Afficher les propriétés d'un ou de plusieurs objets gérés stockés ou le contenu du contexte en cours
SUPPRIMER	caractère d'effacement	Supprimer un ou plusieurs objets gérés de l'espace de nom ou supprimer un sous-contexte vide
MODIFIER	CHG	Modifier le contexte en cours, ce qui permet à l'utilisateur de traverser l'espace de nom du répertoire n'importe où en dessous du contexte initial (en attente d'une autorisation de sécurité)
COPIER	programme de contrôle	Faire une copie d'un objet géré stocké, en le stockant sous un autre nom
DEPLACER	mégavolt	Modifier le nom sous lequel un objet géré est stocké
FIN		Fermer l'outil d'administration

Procédure

- Si l'outil d'administration n'est pas déjà démarré, démarrez-le comme décrit dans «[Démarrage des outils JMSAdmin et JMS30Admin](#)», à la page 757.

L'invite de commande s'affiche, indiquant que l'outil est prêt à accepter les commandes d'administration. Cette invite apparaît initialement comme suit:

```
InitCtx>
```

Pour modifier le contexte en cours, utilisez l'instruction CHANGE comme décrit dans «[Configuration de sous-contextes](#)», à la page 759.

- Entrez les commandes au format suivant:

```
verb [param]*
```

où **verb** est l'un des verbes d'administration répertoriés dans Tableau 38, à la page 758. Toutes les commandes valides contiennent une instruction, qui apparaît au début de la commande sous sa forme standard ou abrégée. Les noms d'instruction ne sont pas sensibles à la casse.

- Pour mettre fin à une commande, appuyez sur Entrée, sauf si vous souhaitez entrer plusieurs commandes ensemble, auquel cas tapez le signe plus (+) directement avant d'appuyer sur Entrée. Généralement, pour arrêter les commandes, appuyez sur Entrée. Toutefois, vous pouvez le remplacer en tapant le signe plus (+) directement avant d'appuyer sur Entrée. Cela vous permet d'entrer des commandes multilignes, comme illustré dans l'exemple suivant:

```
DEFINE Q(BookingsInputQueue) +
QMGR(QM.POLARIS.TEST) +
QUEUE(BOOKINGS.INPUT.QUEUE) +
PORT(1415) +
CCSID(437)
```

- Pour fermer l'outil d'administration, utilisez l'instruction **END**. Cette instruction ne peut pas prendre de paramètres.

Configuration de sous-contextes

Vous pouvez utiliser les instructions **CHANGE**, **DEFINE**, **DISPLAY** et **DELETE** pour configurer des sous-contextes d'espace de nom d'annuaire.

Pourquoi et quand exécuter cette tâche

L'utilisation de ces verbes est décrite dans le tableau suivant.

Syntaxe de la commande	Description
DEFINE CTX (ctxName)	Tente de créer un sous-contexte enfant du contexte en cours, nommé ctxName. Échoue en cas de violation de sécurité, si le sous-contexte existe déjà ou si le nom fourni n'est pas valide.
DISPLAY CTX	Affiche le contenu du contexte en cours. Les objets gérés sont annotés avec a, les sous-contextes avec [D]. Le type Java de chaque objet est également affiché.
DELETE CTX (ctxName)	Tente de supprimer le contexte enfant du contexte en cours portant le nom ctxName. Échoue si le contexte est introuvable, s'il n'est pas vide ou s'il existe une violation de sécurité.
CHANGE CTX (ctxName)	Modifie le contexte en cours, de sorte qu'il fait désormais référence au contexte enfant portant le nom ctxName. L'une des deux valeurs spéciales de ctxName peut être fournie: = ACTIF se déplace vers le parent du contexte en cours = INIT se déplace directement dans le contexte initial Échoue si le contexte spécifié n'existe pas ou s'il existe une violation de sécurité.

Les noms des objets JMS ou Jakarta Messaging stockés dans un environnement LDAP doivent être conformes aux conventions de dénomination LDAP. L'une de ces conventions est que les noms d'objet et de contexte doivent inclure un préfixe, tel que cn= (nom usuel) ou ou= (unité organisationnelle). L'outil d'administration simplifie l'utilisation des fournisseurs de services LDAP en vous permettant de faire référence à des noms d'objet et de contexte sans préfixe. Si vous n'indiquez pas de préfixe, l'outil ajoute automatiquement un préfixe par défaut au nom que vous indiquez. Pour LDAP, il s'agit de cn=. Si nécessaire, vous pouvez modifier le préfixe par défaut en définissant la propriété **NAME_PREFIX** dans le fichier de configuration. Pour plus d'informations, voir «[Configuration des outils JMSAdmin et JMS30Admin](#)», à la page 755.

Remarque : Vous devrez peut-être configurer votre serveur LDAP pour stocker des objets Java . Pour plus d'informations, voir la documentation de votre serveur LDAP.

Création d'objets JMS

Pour créer des objets de destination et de fabrique de connexions JMS ou Jakarta Messaging et les stocker dans un espace de nom JNDI, utilisez le verbe DEFINE . Pour stocker vos objets dans un environnement LDAP, vous devez leur attribuer des noms conformes à certaines conventions. L'outil d'administration peut vous aider à respecter les conventions de dénomination LDAP en ajoutant un préfixe par défaut aux noms d'objet.

Pourquoi et quand exécuter cette tâche

L'instruction DEFINE crée un objet géré avec le type, le nom et les propriétés que vous spécifiez. Le nouvel objet est stocké dans le contexte en cours.

Les noms des objets JMS ou Jakarta Messaging stockés dans un environnement LDAP doivent être conformes aux conventions de dénomination LDAP. L'une de ces conventions est que les noms d'objet et de contexte doivent inclure un préfixe, tel que cn= (nom usuel) ou ou= (unité organisationnelle). L'outil d'administration simplifie l'utilisation des fournisseurs de services LDAP en vous permettant de

faire référence à des noms d'objet et de contexte sans préfixe. Si vous n'indiquez pas de préfixe, l'outil ajoute automatiquement un préfixe par défaut au nom que vous indiquez. Pour LDAP, il s'agit de `cn=`. Si nécessaire, vous pouvez modifier le préfixe par défaut en définissant la propriété **NAME_PREFIX** dans le fichier de configuration. Pour plus d'informations, voir «[Configuration des outils JMSAdmin et JMS30Admin](#)», à la page 755.

Remarque : Vous devrez peut-être configurer votre serveur LDAP pour stocker des objets Java . Pour plus d'informations, voir la documentation de votre serveur LDAP.

Procédure

1. Si l'outil d'administration n'est pas déjà démarré, démarrez-le comme décrit dans «[Démarrage des outils JMSAdmin et JMS30Admin](#)», à la page 757.

L'invite de commande s'affiche, indiquant que l'outil est prêt à accepter les commandes d'administration.

2. Assurez-vous que l'invite de commande affiche le contexte dans lequel vous souhaitez créer le nouvel objet.

Lorsque vous démarrez l'outil d'administration, l'invite apparaît initialement comme suit:

```
InitCtx>
```

Pour modifier le contexte en cours, utilisez l'instruction `CHANGE` comme décrit dans «[Configuration de sous-contextes](#)», à la page 759.

3. Pour créer une fabrique de connexions, une destination de file d'attente ou une destination de rubrique, utilisez la syntaxe de commande suivante:

```
DEFINE TYPE (name) [property]*
```

Autrement dit, entrez le verbe `DEFINE` , suivi d'une `TYPE (name)` référence d'objet géré, suivi de zéro ou plusieurs *propriétés* (voir [Propriétés des objets IBM MQ classes for JMS](#)).

4. Pour créer une fabrique de connexions, une destination de file d'attente ou une destination de rubrique, utilisez la syntaxe de commande suivante:

```
DEFINE TYPE (name) [property]*
```

5. Pour afficher l'objet nouvellement créé, utilisez le verbe `DISPLAY` avec la syntaxe de commande suivante:

```
DISPLAY TYPE (name)
```

Exemple

L'exemple suivant illustre une file d'attente appelée `testQueue` créée dans le contexte initial à l'aide de l'instruction `DEFINE` . Etant donné que cet objet est stocké dans un environnement LDAP, bien que le nom d'objet `testQueue` ne soit pas entré avec un préfixe, l'outil en ajoute automatiquement un pour garantir la conformité avec la convention de dénomination LDAP. La soumission de la commande `DISPLAY Q(testQueue)` entraîne également l'ajout de ce préfixe.

JM 3.0

```
InitCtx> DEFINE Q(testQueue)
```

```
InitCtx> DISPLAY CTX
```

```
Contents of InitCtx
```

```
a cn=testQueue          com.ibm.mq.jakarta.jms.MQQueue
```

```
1 Object(s)
```

```
0 Context(s)
1 Binding(s), 1 Administered
```

JMS 2.0

```
InitCtx> DEFINE Q(testQueue)

InitCtx> DISPLAY CTX

Contents of InitCtx

a cn=testQueue          com.ibm.mq.jms.MQQueue

1 Object(s)
0 Context(s)
1 Binding(s), 1 Administered
```

Exemple de conditions d'erreur lors de la création d'un objet JMS

Un certain nombre de conditions d'erreur communes peuvent se produire lorsque vous créez un objet.

Voici des exemples de ces conditions d'erreur:

CipherSpec mappé à CipherSuite

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) SSLCIPHERSUITE(RC4_MD5_US)
WARNING: Converting CipherSpec RC4_MD5_US to
CipherSuite SSL_RSA_WITH_RC4_128_MD5
```

Propriété non valide pour l'objet

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) PRIORITY(4)
Unable to create a valid object, please check the parameters supplied
Invalid property for a QCF: PRI
```

Type non valide pour la valeur de propriété

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) CCSID(english)
Unable to create a valid object, please check the parameters supplied
Invalid value for CCS property: English
```

Conflit de propriétés-client/bin

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) HOSTNAME(polaris.hursley.ibm.com)
Unable to create a valid object, please check the parameters supplied
Invalid property in this context: Client-bindings attribute clash
```

Conflit de propriétés-Initialisation de l'exit

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) SECEXITINIT(initStr)
Unable to create a valid object, please check the parameters supplied
Invalid property in this context: ExitInit string supplied
without Exit string
```

Valeur de propriété en dehors de la plage valide

```
InitCtx/cn=Trash> DEFINE Q(testQ) PRIORITY(12)
Unable to create a valid object, please check the parameters supplied
Invalid value for PRI property: 12
```

propriété inconnue

```
InitCtx/cn=Trash> DEFINE QCF(testQCF) PIZZA(ham and mushroom)
Unable to create a valid object, please check the parameters supplied
Unknown property: PIZZA
```

Voici des exemples de conditions d'erreur qui peuvent survenir sur Windows lors de la recherche d'objets gérés JNDI à partir d'une application JMS .

1. Si vous utilisez le fournisseur JNDI WebSphere , `com.ibm.websphere.naming.WsnInitialContextFactory`, vous devez utiliser une barre oblique (/) pour accéder aux objets gérés définis dans les sous-contextes, par exemple, `jms /MyQueueName`. Si vous utilisez une barre oblique inversée (\), une exception `InvalidName` est émise.
2. Si vous utilisez le fournisseur JNDI Oracle , `com.sun.jndi.fscontext.RefFSContextFactory`, vous devez utiliser une barre oblique inversée (\) pour accéder aux objets gérés définis dans les sous-contextes, par exemple, `ctx1\\fred`. Si vous utilisez une barre oblique (/), une exception `NameNotFoundException` est émise.

Configuration des objets JMS

Vous pouvez utiliser les instructions ALTER, DEFINE, DISPLAY, DELETE, COPY et MOVE pour manipuler les objets gérés dans l'espace de nom du répertoire.

Pourquoi et quand exécuter cette tâche

Le Tableau 40, à la page 763 récapitule l'utilisation de ces verbes. Remplacez *TYPE* par le mot clé qui représente l'objet géré requis, comme décrit dans «[Configuration des objets JMS et Jakarta Messaging à l'aide des outils d'administration](#)», à la page 753.

<i>Tableau 40. Syntaxe et description des commandes utilisées pour manipuler les objets gérés</i>	
Syntaxe de la commande	Description
ALTER <i>TYPE</i> (nom) [propriété] *	Tente de mettre à jour les propriétés de l'objet géré avec celles fournies. Échoue en cas de violation de sécurité, si l'objet spécifié est introuvable ou si les nouvelles propriétés fournies ne sont pas valides.
DEFINE <i>TYPE</i> (nom) [propriété] *	Tente de créer un objet géré de type <i>TYPE</i> avec les propriétés fournies et le stocke sous le nom <i>name</i> dans le contexte en cours. Echec en cas de violation de sécurité, si le nom fourni n'est pas valide ou qu'un objet de ce nom existe, ou si les propriétés fournies ne sont pas valides.
DISPLAY <i>TYPE</i> (nom)	Affiche les propriétés de l'objet géré de type <i>TYPE</i> , lié sous le nom <i>name</i> dans le contexte en cours. Échoue si l'objet n'existe pas ou s'il existe une violation de sécurité.
DELETE <i>TYPE</i> (nom)	Tente de supprimer l'objet géré de type <i>TYPE</i> , portant le nom <i>name</i> , du contexte en cours. Échoue si l'objet n'existe pas ou s'il existe une violation de sécurité.
COPY <i>TYPE</i> (nameA) <i>TYPE</i> (nameB)	Effectue une copie de l'objet géré de type <i>TYPE</i> , portant le nom <i>nameA</i> , en nommant la copie <i>nameB</i> . Tout cela se produit dans la portée du contexte en cours. Echec si l'objet à copier n'existe pas, s'il existe un objet nommé <i>nameB</i> ou s'il existe une violation de sécurité.
MOVE <i>TYPE</i> (nameA) <i>TYPE</i> (nameB)	Déplace (renomme) l'objet géré de type <i>TYPE</i> , portant le nom <i>nameA</i> , vers <i>nameB</i> . Tout cela se produit dans la portée du contexte en cours. Échoue si l'objet à déplacer n'existe pas, s'il existe un objet nommé <i>nameB</i> ou s'il existe une violation de sécurité.

JMS 2.0 Configuration des ressources JMS 2.0 dans WebSphere Application Server

Pour configurer des ressources JMS 2.0 dans WebSphere Application Server, vous pouvez utiliser la console d'administration ou les commandes wsadmin.

Avant de commencer

JM 3.0 Bien que IBM MQ 9.3 et les versions ultérieures prennent en charge [Jakarta Messaging 3.0](#), WebSphere Application Server ne dispose pas actuellement d'une prise en charge équivalente. Par conséquent, dans WebSphere Application Server, vous configurez les ressources Java Message Service 2.0 .

Pourquoi et quand exécuter cette tâche

Les applications Java Message Service 2.0 s'appuient généralement sur des objets configurés en externe qui décrivent comment l'application se connecte à son fournisseur JMS et aux destinations auxquelles elle accède. Les applications JMS utilisent Java Naming Directory Interface (JNDI) pour accéder aux types d'objet suivants lors de l'exécution:

- Spécifications d'activation (utilisées par les serveurs d'applications Java EE)
- Fabriques de connexions unifiées (avec JMS 1.1 et versions ultérieures, les fabriques de connexions indépendantes du domaine (unifiées) sont préférées aux fabriques de connexions de file d'attente spécifiques au domaine et aux fabriques de connexions de rubriques)
- Fabriques de connexions de sujet (utilisées par les applications JMS 1.0)
- Fabriques de connexions de file d'attente (utilisées par les applications JMS 1.0)
- Files d'attente
- Rubriques

Grâce au fournisseur de messagerie IBM MQ de WebSphere Application Server, les applications de messagerie Java Message Service (JMS) peuvent utiliser votre système IBM MQ comme fournisseur externe de ressources de messagerie JMS. Pour activer cette approche, vous configurez le fournisseur de messagerie IBM MQ dans WebSphere Application Server afin de définir les ressources JMS pour la connexion à n'importe quel gestionnaire de files d'attente sur le réseau IBM MQ .

Vous pouvez utiliser WebSphere Application Server pour configurer les ressources IBM MQ pour les applications (par exemple, les fabriques de connexions de file d'attente) et pour gérer les messages et les abonnements associés aux destinations JMS . Vous administrez la sécurité via IBM MQ.

Tâches associées

[Utilisation conjointe de IBM MQ et de WebSphere Application Server](#)

WebSphere Application Server rubriques

[Interopération à l'aide du fournisseur de messagerie IBM MQ](#)

[Gestion de la messagerie avec le fournisseur de messagerie IBM MQ](#)

[Mappage des noms de panneau de la console d'administration aux noms de commande et aux noms IBM MQ](#)

JMS 2.0 Configuration des ressources JMS 2.0 à l'aide de la console d'administration

Vous pouvez utiliser la console d'administration WebSphere Application Server pour configurer des spécifications d'activation, des fabriques de connexions et des destinations pour le fournisseur IBM MQ JMS .

Pourquoi et quand exécuter cette tâche

Vous pouvez utiliser la console d'administration WebSphere Application Server pour créer, afficher ou modifier l'une des ressources suivantes:

- Spécifications d'activation
- Fabriques de connexions indépendantes du domaine (JMS 1.1 ou version ultérieure)
- Fabriques de connexions de file d'attente
- Fabriques de connexions de sujet
- Files d'attente
- Rubriques

Les étapes suivantes présentent les manières dont vous pouvez utiliser la console d'administration pour configurer des ressources JMS à utiliser avec le fournisseur de messagerie IBM MQ . Chaque étape inclut le nom de la rubrique dans la documentation du produit WebSphere Application Server à laquelle vous pouvez vous référer pour plus d'informations. Pour obtenir des liens vers ces rubriques dans le IBM Documentation, voir *Liens connexes* .

Dans une cellule WebSphere Application Server de version mixte, vous pouvez administrer les ressources IBM MQ sur les noeuds de toutes les versions. Cependant, certaines propriétés ne sont pas disponibles sur toutes les versions. Dans ce cas, seules les propriétés de ce noeud particulier sont affichées dans la console d'administration.

Procédure

Pour créer ou configurer une spécification d'activation à utiliser avec le fournisseur de messagerie IBM MQ :

- Pour créer une spécification d'activation, utilisez l'assistant de création de ressource IBM MQ JMS . Vous pouvez utiliser l'assistant pour spécifier tous les détails de la spécification d'activation ou vous pouvez choisir de spécifier les détails de connexion pour IBM MQ à l'aide d'une table de définition de canal du client (CCDT). Lorsque vous spécifiez les détails de connexion à l'aide de l'assistant, vous pouvez choisir d'entrer les informations d'hôte et de port séparément ou, si vous utilisez un gestionnaire de files d'attente multi-instance, d'entrer les informations d'hôte et de port sous la forme d'une liste de noms de connexion. Pour plus d'informations, voir *Création d'une spécification d'activation pour le fournisseur de messagerie IBM MQ*.
- Pour afficher ou modifier les propriétés de configuration d'une spécification d'activation, utilisez le panneau des paramètres de la fabrique de connexions du fournisseur de messagerie IBM MQ de la console d'administration.

Ces propriétés de configuration contrôlent la façon dont les connexions sont créées pour les files d'attente et les rubriques associées. Pour plus d'informations, voir *Configuration d'une spécification d'activation pour le fournisseur de messagerie IBM MQ*.

Pour créer ou configurer une fabrique de connexions unifiée, une fabrique de connexions de file d'attente ou une fabrique de connexions de rubriques à utiliser avec le fournisseur de messagerie IBM MQ :

- Pour créer une fabrique de connexions, sélectionnez d'abord le type de fabrique de connexions que vous souhaitez créer, puis utilisez l'assistant de création de ressources IBM MQ JMS pour spécifier les détails.
 - Si votre application JMS est destinée à utiliser uniquement la messagerie point-à-point, créez une fabrique de connexions spécifique au domaine pour le domaine de messagerie point-à-point qui peut être utilisée pour créer des connexions spécifiquement pour la messagerie point-à-point.
 - Si votre application JMS est destinée uniquement à utiliser la messagerie de publication / abonnement, créez une fabrique de connexions spécifique au domaine de messagerie de publication / abonnement qui peut être utilisée pour créer des connexions spécifiques à la messagerie de publication / abonnement.
 - Pour JMS 1.1 ou version ultérieure, créez une fabrique de connexions indépendante du domaine qui peut être utilisée à la fois pour la messagerie point-à-point et la messagerie de publication /

abonnement, ce qui permet à votre application d'effectuer à la fois le travail point-à-point et le travail de publication / abonnement sous la même transaction.

Vous pouvez choisir d'utiliser l'assistant pour spécifier tous les détails de la fabrique de connexions ou de spécifier les détails de connexion pour IBM MQ à l'aide d'une table de définition de canal du client (CCDT). Lorsque vous spécifiez les détails de connexion à l'aide de l'assistant, vous pouvez choisir d'entrer les informations d'hôte et de port séparément ou, si vous utilisez un gestionnaire de files d'attente multi-instance, d'entrer les informations d'hôte et de port sous la forme d'une liste de noms de connexion. Pour plus d'informations, voir *Création d'une fabrique de connexions pour le fournisseur de messagerie IBM MQ*.

Pour afficher ou modifier les propriétés de configuration d'une fabrique de connexions:

- Utilisez le panneau des paramètres de la fabrique de connexions de la console d'administration pour le type de fabrique de connexions que vous souhaitez configurer.

Les propriétés de configuration contrôlent la façon dont les connexions sont créées pour les files d'attente et les rubriques associées. Pour plus d'informations, voir *Configuration d'une fabrique de collecte pour le fournisseur de messagerie IBM MQ*, *Configuration d'une fabrique de collecte de files d'attente pour le fournisseur de messagerie IBM MQ* ou *Configuration d'une fabrique de collecte de rubriques pour le fournisseur de messagerie IBM MQ*.

Pour configurer une destination de file d'attente JMS pour la messagerie point-à-point avec le fournisseur de messagerie IBM MQ :

- Utilisez le panneau des paramètres de file d'attente du fournisseur de messagerie IBM MQ de la console d'administration pour définir les types de propriété suivants:
 - Propriétés générales, y compris les propriétés d'administration et de file d'attente IBM MQ .
 - Propriétés de connexion qui indiquent comment se connecter au gestionnaire de files d'attente qui héberge la file d'attente.
 - Propriétés avancées qui contrôlent le comportement des connexions établies avec les destinations du fournisseur de messagerie IBM MQ .
 - Toutes les propriétés personnalisées de la destination de file d'attente.

Pour plus d'informations, voir *Configuration d'une file d'attente pour le fournisseur de messagerie IBM MQ*.

Pour créer ou configurer une destination de rubrique JMS pour la messagerie de publication / abonnement avec le fournisseur de messagerie IBM MQ :

- Le panneau des paramètres de rubrique du fournisseur de messagerie IBM MQ permet de définir les types de propriété suivants:
 - Propriétés générales, y compris les propriétés d'administration et de rubrique IBM MQ .
 - Propriétés avancées qui contrôlent le comportement des connexions établies avec les destinations du fournisseur de messagerie IBM MQ .
 - Toutes les propriétés personnalisées de la destination de file d'attente.

Pour plus d'informations, voir *Configuration d'une rubrique pour le fournisseur de messagerie IBM MQ*.

Concepts associés

«Gestionnaires de files d'attente multi-instance», à la page 547

Les gestionnaires de files d'attente multi-instance sont des instances du même gestionnaire de files d'attente configurées sur des serveurs différents. Une instance du gestionnaire de files d'attente est définie en tant qu'instance active et une autre instance est définie en tant qu'instance de secours. En cas de défaillance de l'instance active, le gestionnaire de files d'attente multi-instance redémarre automatiquement sur le serveur de secours.

Tâches associées

«Configuration d'une table de définition de canal du client au format binaire», à la page 45

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut

est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

«Configuration de la messagerie de type publication/abonnement», à la page 468

Vous pouvez démarrer, arrêter et afficher le statut de la publication / abonnement en file d'attente.

Vous pouvez également ajouter et supprimer des flux et ajouter et supprimer des gestionnaires de files d'attente à partir d'une hiérarchie de courtiers.

WebSphere Application Server rubriques

[Spécifications d'activation du fournisseur de messagerie IBM MQ](#)

[Création d'une spécification d'activation pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une spécification d'activation pour le fournisseur de messagerie IBM MQ](#)

[Création d'une fabrique de connexions pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une fabrique de connexions unifiée pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une fabrique de connexions de file d'attente pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une fabrique de connexions de rubriques pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une file d'attente pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une rubrique pour le fournisseur de messagerie IBM MQ](#)

JMS 2.0 Configuration des ressources JMS 2.0 à l'aide des commandes de scriptage wsadmin

Vous pouvez utiliser les commandes de scriptage wsadmin WebSphere Application Server pour créer, modifier, supprimer ou afficher des informations sur les spécifications d'activation, les fabriques de connexions, les files d'attente et les rubriques JMS . Vous pouvez également afficher et gérer les paramètres de l'adaptateur de ressources IBM MQ .

Pourquoi et quand exécuter cette tâche

Les étapes suivantes présentent les manières dont vous pouvez utiliser les commandes wsadmin WebSphere Application Server pour configurer des ressources JMS à utiliser avec le fournisseur de messagerie IBM MQ . Pour plus d'informations sur l'utilisation de ces commandes, voir *Liens associés* pour obtenir des liens vers la documentation du produit WebSphere Application Server .

Pour exécuter une commande, utilisez l'objet AdminTask du client de scriptage wsadmin.

Après avoir utilisé une commande pour créer un objet ou effectuer des modifications, sauvegardez vos modifications dans la configuration principale. Par exemple, utilisez la commande suivante :

```
AdminConfig.save()
```

Pour afficher la liste des commandes d'administration du fournisseur de messagerie IBM MQ disponibles, ainsi qu'une brève description de chaque commande, entrez la commande suivante à l'invite wsadmin:

```
print AdminTask.help('WMQAdminCommands')
```

Pour afficher l'aide générale sur une commande donnée, entrez la commande suivante à l'invite wsadmin:

```
print AdminTask.help('command_name')
```

Procédure

Pour répertorier toutes les ressources du fournisseur de messagerie IBM MQ définies dans la portée à laquelle une commande est émise, utilisez les commandes suivantes.

- Pour répertorier les spécifications d'activation, utilisez la commande **listWMQActivationSpecs** .

- Pour répertorier les fabriques de connexions, utilisez la commande **listWMQConnectionFactories** .
 - Pour répertorier les destinations de type file d'attente, utilisez la commande **listWMQQueues** .
 - Pour répertorier les destinations de type rubrique, utilisez la commande **listWMQTopics** .
- Pour créer une ressource JMS pour le fournisseur de messagerie IBM MQ à une portée spécifique, utilisez les commandes suivantes.

- Pour créer une spécification d'activation, utilisez la commande **createWMQActivationSpec** .
Vous pouvez soit créer une spécification d'activation en spécifiant tous les paramètres à utiliser pour établir une connexion, soit créer la spécification d'activation afin qu'elle utilise une table de définition de canal du client (CCDT) pour localiser le gestionnaire de files d'attente auquel se connecter.
- Pour créer une fabrique de connexions, utilisez la commande **createWMQConnectionFactory** en utilisant le paramètre **-type** pour spécifier le type de fabrique de connexions que vous souhaitez créer:
 - Si votre application JMS est destinée à utiliser uniquement la messagerie point-à-point, créez une fabrique de connexions spécifique au domaine pour le domaine de messagerie point-à-point qui peut être utilisée pour créer des connexions spécifiquement pour la messagerie point-à-point.
 - Si votre application JMS est destinée uniquement à utiliser la messagerie de publication / abonnement, créez une fabrique de connexions spécifique au domaine de messagerie de publication / abonnement qui peut être utilisée pour créer des connexions spécifiques à la messagerie de publication / abonnement.
 - Pour JMS 1.1 ou version ultérieure, créez une fabrique de connexions indépendante du domaine qui peut être utilisée à la fois pour la messagerie point-à-point et la messagerie de publication / abonnement, ce qui permet à votre application d'effectuer à la fois le travail point-à-point et le travail de publication / abonnement sous la même transaction.

Le type par défaut est la fabrique de connexions indépendante du domaine.

- Pour créer une destination de type file d'attente, utilisez la commande **createWMQQueue** .
 - Pour créer une destination de type rubrique, utilisez la commande **createWMQTopic** .
- Pour modifier une ressource JMS pour le fournisseur de messagerie IBM MQ sur une portée spécifique, utilisez les commandes suivantes.
- Pour modifier une spécification d'activation, utilisez la commande **modifyWMQActivationSpec** .
Vous ne pouvez pas modifier le type d'une spécification d'activation. Par exemple, vous ne pouvez pas créer de spécification d'activation dans laquelle vous entrez toutes les informations de configuration manuellement, puis les modifier pour utiliser une table de définition de canal du client.
 - Pour modifier une fabrique de connexions, utilisez la commande **modifyWMQConnectionFactory** .
 - Pour modifier une destination de type file d'attente, utilisez la commande **modifyWMQQueue** .
 - Pour modifier une destination de type rubrique, utilisez la commande **modifyWMQTopic** .
- Pour supprimer une ressource JMS pour le fournisseur de messagerie IBM MQ sur une portée spécifique, utilisez les commandes suivantes.
- Pour supprimer une spécification d'activation, utilisez la commande **deleteWMQActivationSpec** .
 - Pour supprimer une fabrique de connexions, utilisez la commande **deleteWMQConnectionFactory** .
 - Pour supprimer une destination de type file d'attente, utilisez la commande **deleteWMQQueue** .
 - Pour supprimer une destination de type rubrique, utilisez la commande **deleteWMQTopic** .
- Pour afficher des informations sur une ressource de fournisseur de messagerie IBM MQ spécifique, utilisez les commandes suivantes.
- Pour afficher tous les paramètres et leurs valeurs associés à une spécification d'activation particulière, utilisez la commande **showWMQActivationSpec** .
 - Pour afficher tous les paramètres et leurs valeurs associés à une fabrique de connexions particulière, utilisez la commande **showWMQConnectionFactory** .

- Pour afficher tous les paramètres et leurs valeurs associés à une destination de type de file d'attente particulière, utilisez la commande **showWMQQueue** .
- Pour afficher tous les paramètres et leurs valeurs associés à une destination de type rubrique, utilisez la commande **deleteWMQTopic** .

Pour gérer les paramètres de l'adaptateur de ressources IBM MQ ou du fournisseur de messagerie IBM MQ , utilisez les commandes suivantes.

- Pour gérer les paramètres de l'adaptateur de ressources IBM MQ installé sur une portée particulière, utilisez la commande **manageWMQ** .
- Pour afficher tous les paramètres et leurs valeurs pouvant être définis par la commande **manageWMQ** , utilisez la commande **showWMQ** . Ces paramètres sont liés à l'adaptateur de ressources IBM MQ ou au fournisseur de messagerie IBM MQ . La commande **showWMQ** affiche également les propriétés personnalisées définies sur l'adaptateur de ressources IBM MQ .

Concepts associés

[«Gestionnaires de files d'attente multi-instance», à la page 547](#)

Les gestionnaires de files d'attente multi-instance sont des instances du même gestionnaire de files d'attente configurées sur des serveurs différents. Une instance du gestionnaire de files d'attente est définie en tant qu'instance active et une autre instance est définie en tant qu'instance de secours. En cas de défaillance de l'instance active, le gestionnaire de files d'attente multi-instance redémarre automatiquement sur le serveur de secours.

Tâches associées

[«Configuration d'une table de définition de canal du client au format binaire», à la page 45](#)

La table de définition de canal du client (CCDT) détermine les définitions de canal et les informations d'authentification utilisées par les applications client pour se connecter au gestionnaire de files d'attente. Sur Multiplatforms, une table de définition de canal du client binaire contenant les paramètres par défaut est créée automatiquement lors de la création du gestionnaire de files d'attente. La commande **runmqsc** permet de mettre à jour une table de définition de canal du client binaire.

[«Configuration de la messagerie de type publication/abonnement», à la page 468](#)

Vous pouvez démarrer, arrêter et afficher le statut de la publication / abonnement en file d'attente.

Vous pouvez également ajouter et supprimer des flux et ajouter et supprimer des gestionnaires de files d'attente à partir d'une hiérarchie de courtiers.

WebSphere Application Server rubriques

commande [**createWMQActivationSpec**](#)

commande [**createWMQConnectionFactory**](#)

commande [**createWMQQueue**](#)

commande [**createWMQTopic**](#)

commande [**deleteWMQActivationSpec**](#)

commande [**deleteWMQConnectionFactory**](#)

commande [**deleteWMQQueue**](#)

commande [**deleteWMQTopic**](#)

commande [**listWMQActivationSpecs**](#)

commande [**listWMQConnectionFactories**](#)

commande [**listWMQQueues**](#)

commande [**listWMQTopics**](#)

commande [**modifyWMQActivationSpec**](#)

commande [**modifyWMQConnectionFactory**](#)

commande [**modifyWMQQueue**](#)

commande [**modifyWMQTopic**](#)

commande [**showWMQActivationSpec**](#)

commande [**showWMQConnectionFactory**](#)

commande [**showWMQQueue**](#)

commande [**showWMQTopic**](#)

commande **showWMQ**

commande **manageWMQ**

JMS 2.0 Utilisation d'abonnements partagés JMS 2.0

Dans WebSphere Application Server traditional 9.0, vous pouvez configurer et utiliser des abonnements partagés JMS 2.0 avec IBM MQ 9.0.

Pourquoi et quand exécuter cette tâche

La spécification JMS 2.0 a introduit le concept d'abonnements partagés, qui permet à un abonnement unique d'être ouvert par un ou plusieurs consommateurs. Les messages sont partagés entre tous ces consommateurs. Il n'y a pas de restriction lorsque ces consommateurs sont aussi longtemps qu'ils se connectent au même gestionnaire de files d'attente.

Les abonnements partagés peuvent être durables ou non durables, avec la même sémantique que les abonnements non partagés.

Pour qu'un consommateur puisse identifier l'abonnement à utiliser, il doit fournir un nom d'abonnement. Cela est similaire aux abonnements durables non partagés, mais un nom d'abonnement est requis dans tous les cas où un abonnement partagé est requis. Toutefois, un clientID n'est pas requis dans le cas d'un abonnement partagé durable ; vous pouvez en indiquer un, mais il n'est pas obligatoire.

Bien que les abonnements partagés puissent être considérés comme un mécanisme d'équilibrage de charge, ni dans IBM MQ ni dans la spécification JMS 2.0, il n'y a aucun engagement quant à la façon dont les messages sont distribués entre les consommateurs.

Dans WebSphere Application Server traditional 9.0, un adaptateur de ressources IBM MQ 9.0 est préinstallé.

Les étapes suivantes montrent comment configurer une spécification d'activation pour utiliser un abonnement durable partagé ou un abonnement non durable partagé à l'aide de la console d'administration WebSphere Application Server traditional.

Procédure

Créez d'abord les objets dans JNDI.

1. Créez une destination de rubrique dans JNDI normalement (voir [«Configuration des ressources JMS 2.0 à l'aide de la console d'administration»](#), à la page 764).
2. Créez la spécification d'activation (voir [«Configuration des ressources JMS 2.0 à l'aide de la console d'administration»](#), à la page 764).

Vous pouvez créer la spécification d'activation avec les propriétés dont vous avez besoin. Si vous souhaitez utiliser un abonnement durable, vous pouvez le sélectionner lors de la création et spécifier un nom. Si vous souhaitez utiliser un abonnement non durable, vous ne pouvez pas spécifier de nom à ce stade. A la place, vous devez créer une propriété personnalisée pour le nom de l'abonnement.

Mettez à jour la spécification d'activation que vous avez créée avec les propriétés personnalisées requises. Vous devrez peut-être spécifier deux propriétés personnalisées:

- Dans tous les cas, vous devez créer une propriété personnalisée pour indiquer que cette spécification d'activation doit utiliser un abonnement partagé.
- Si l'abonnement a été créé en tant que non durable, la propriété de nom d'abonnement doit être définie en tant que propriété personnalisée.

Le tableau suivant présente la valeur valide que vous pouvez spécifier pour chaque propriété personnalisée:

Nom de la propriété	Tapez	Valeur valides
sharedSubscription	String	true, false

Nom de la propriété	Tapez	Valeur valides
subscriptionName	String	Chaîne Java de longueur différente de zéro

3. Sélectionnez la spécification d'activation dans la liste affichée dans le formulaire **Collection de spécifications d'activation** .

Les détails de la spécification d'activation sont affichés dans le formulaire **IBM MQ** .

4. Dans le formulaire **IBM MQ** , cliquez sur **Propriétés personnalisées**.

Le formulaire **Propriétés personnalisées** s'affiche.

5. Si vous utilisez un abonnement non durable, créez la propriété personnalisée subscriptionName .

Dans le panneau **Propriétés personnalisées** de la spécification d'activation, cliquez sur **Nouveau**, puis entrez les détails suivants:

Nom

Nom de la propriété personnalisée, qui dans ce cas est subscriptionName.

Valeur

Valeur de la propriété personnalisée. Vous pouvez utiliser les noms JNDI dans la zone **Valeur** , par exemple WASSharedSubOne.

Tapez

Type de la propriété personnalisée. Sélectionnez le type de propriété personnalisée dans la liste, qui dans ce cas doit être `java.lang.String`.

6. Pour un abonnement durable partagé et un abonnement non durable partagé, créez la propriété personnalisée sharedSubscription .

Dans le panneau **Propriétés personnalisées** de la spécification d'activation, cliquez sur **Nouveau**, puis entrez les détails suivants:

Nom

Nom de la propriété personnalisée, qui dans ce cas est sharedSubscription.

Valeur

Valeur de la propriété personnalisée. Pour indiquer que la spécification d'activation utilise un abonnement partagé, définissez la valeur sur `true`. Si vous souhaitez par la suite arrêter d'utiliser un abonnement partagé pour cette spécification d'activation, vous pouvez le faire en définissant la valeur de cette propriété personnalisée sur `false`.

Tapez

Type de la propriété personnalisée. Sélectionnez le type de propriété personnalisée dans la liste, qui dans ce cas doit être `java.lang.String`.

7. Une fois les propriétés définies, redémarrez le serveur d'applications.

Les beans gérés par message (MDB) des spécifications d'activation sont ensuite gérés à l'arrivée des messages, mais seuls les beans gérés par message partagent les messages envoyés.

Concepts associés

[Abonnements clonés et partagés](#)

[Durabilité des abonnements](#)

Tâches associées

[Configuration de l'adaptateur de ressources pour les communications entrantes](#)

Informations liées à WebSphere Application Server traditional 9.0

[Configuration d'une rubrique pour le fournisseur de messagerie IBM MQ](#)

[Spécifications d'activation du fournisseur de messagerie IBM MQ](#)

[Création d'une spécification d'activation pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une spécification d'activation pour le fournisseur de messagerie IBM MQ](#)

JMS 2.0 Utilisation des propriétés JMS 2.0 ConnectionFactory et Destination Lookup

Dans WebSphere Application Server traditional 9.0, les propriétés ConnectionFactoryLookup et DestinationLookup d'une spécification d'activation peuvent être fournies avec le nom JNDI d'un objet géré à utiliser de préférence aux autres propriétés de spécification d'activation.

Pourquoi et quand exécuter cette tâche

La spécification JMS 2.0 spécifie deux propriétés supplémentaires sur la spécification d'activation utilisée pour piloter les beans gérés par message (MDB). Auparavant, chaque fournisseur devait spécifier des propriétés personnalisées sur la spécification d'activation pour fournir les détails requis pour se connecter à un système de messagerie et pour définir la destination à partir de laquelle extraire les messages.

Les propriétés connectionFactoryLookup et destinationLookup désormais standard peuvent être utilisées pour donner un nom JNDI à l'objet approprié à rechercher et à utiliser. Dans WebSphere Application Server traditional 9.0, un adaptateur de ressources IBM MQ 9.0 est préinstallé.

Les étapes suivantes montrent comment personnaliser et utiliser ces deux propriétés à l'aide de la console d'administration WebSphere Application Server traditional.

Procédure

Créez d'abord les objets dans JNDI.

1. Créez la ConnectionFactory dans JNDI normalement (voir [«Configuration des ressources JMS 2.0 à l'aide de la console d'administration»](#), à la page 764).
2. Créez la destination dans JNDI normalement (voir [«Configuration des ressources JMS 2.0 à l'aide de la console d'administration»](#), à la page 764).
L'objet Destination doit avoir les valeurs correctes.
3. Créez la spécification d'activation à l'aide des valeurs requises (voir [«Configuration des ressources JMS 2.0 à l'aide de la console d'administration»](#), à la page 764).

Vous pouvez créer la spécification d'activation avec les propriétés dont vous avez besoin. Toutefois, vous devez garder à l'esprit les considérations suivantes:

- Si vous souhaitez que l'adaptateur de ressources IBM MQ utilise les propriétés de recherche de la fabrique de connexions et de la destination Java EE, il est moins pertinent de savoir quelles propriétés sont utilisées lorsque vous créez la spécification d'activation (voir [PropriétésActivationSpec ConnectionFactoryLookup et DestinationLookup](#)).
- Toutefois, toute propriété qui n'est pas déjà définie sur la fabrique de connexions ou sur la destination doit toujours être spécifiée sur la spécification d'activation. Par conséquent, vous devez définir les propriétés du consommateur de connexion et les propriétés supplémentaires, ainsi que les informations d'authentification utilisées lors de la création d'une connexion.
- Parmi les propriétés définies dans la fabrique de connexions, la propriété ClientID a un traitement spécial. En effet, un scénario courant consiste à utiliser une seule fabrique de connexions avec plusieurs spécifications d'activation. Cela simplifie l'administration. Cependant, la spécification JMS requiert des ID client uniques. Par conséquent, la spécification d'activation doit avoir la possibilité de remplacer toute valeur définie dans la ConnectionFactory. Si aucun ClientID n'est défini sur la spécification d'activation, toute valeur de la fabrique de connexions est utilisée.

Mettez à jour la spécification d'activation que vous avez créée avec les deux nouvelles propriétés personnalisées à l'aide de la console d'administration WebSphere Application Server, comme décrit à l'étape «4», à la page 773, ou utilisez des annotations à la place, comme décrit à l'étape «5», à la page 773.

4. Mettez à jour la spécification d'activation dans la console d'administration WebSphere Application Server .

Ces deux propriétés doivent être définies dans le panneau des propriétés personnalisées de la spécification d'activation. Ces propriétés ne sont pas présentes dans les principaux panneaux de spécification d'activation ou dans l'assistant de création de spécification d'activation.

- a) Sélectionnez la spécification d'activation dans la liste affichée dans le formulaire **Collection de spécifications d'activation** .

Les détails de la spécification d'activation sont affichés dans le formulaire **IBM MQ** .

- b) Dans le formulaire **IBM MQ** , cliquez sur **Propriétés personnalisées**.

Le formulaire **Propriétés personnalisées** s'affiche.

- c) Dans le formulaire **Propriétés personnalisées** , créez deux nouvelles propriétés personnalisées, toutes deux de type `java.lang.String`.

Dans chaque cas, cliquez sur **Nouveau** , puis entrez les détails suivants pour la propriété personnalisée:

Nom

Nom de la propriété personnalisée, `connectionFactoryLookup` ou `destinationLookup`.

Valeur

Valeur de la propriété personnalisée. Vous pouvez utiliser les noms JNDI dans la zone **Valeur** , par exemple `QuoteCF` et `QuoteQ`.

Tapez

Type de la propriété personnalisée. Sélectionnez le type de propriété personnalisée dans la liste, qui dans ce cas doit être `java.lang.String`.

Le bean géré par message déployé va maintenant utiliser ces valeurs pour créer la fabrique de connexions et la destination. Lors du déploiement du bean géré par message, il n'est pas nécessaire de définir la configuration de la valeur JNDI .

5. Utilisez des annotations à la place de la spécification d'activation.

Il est possible d'utiliser des annotations dans le code MDB pour spécifier également des valeurs. Par exemple, en utilisant les noms JNDI `QuoteCF` et `QuoteQ`, voici à quoi ressemblerait le code:

```
@MessageDriven(activationConfig = {
    @ActivationConfigProperty(propertyName = "destinationType" , propertyValue =
"javax.jms.Topic" ),
    @ActivationConfigProperty(propertyName = "destinationLookup" , propertyValue =
"QuoteQ" ),
    @ActivationConfigProperty(propertyName = "connectionFactoryLookup" , propertyValue
= "QuoteCF" )}, mappedName = "LookupMDB" )
@TransactionAttribute(TransactionAttributeType.REQUIRED)
@TransactionManagement(TransactionManagementType.CONTAINER)
publicclass LookupMDB implements MessageListener {
```

Tâches associées

[Configuration de l'adaptateur de ressources pour les communications entrantes](#)

Informations liées à WebSphere Application Server traditional 9.0

[Configuration d'une fabrique de connexions unifiée pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une rubrique pour le fournisseur de messagerie IBM MQ](#)

[Spécifications d'activation du fournisseur de messagerie IBM MQ](#)

[Création d'une spécification d'activation pour le fournisseur de messagerie IBM MQ](#)

[Configuration d'une spécification d'activation pour le fournisseur de messagerie IBM MQ](#)

[Configuration des propriétés personnalisées pour les ressources du IBM MQ fournisseur de messagerie JMS](#)

Configuration de WebSphere Application Server pour utiliser le dernier niveau de maintenance de l'adaptateur de ressources

Pour vous assurer que l'adaptateur de ressources IBM MQ est automatiquement mis à jour vers le dernier niveau de maintenance disponible lorsque vous appliquez des groupes de correctifs WebSphere Application Server, vous pouvez configurer tous les serveurs de votre environnement pour qu'ils utilisent la dernière version de l'adaptateur de ressources contenue dans le groupe de correctifs WebSphere Application Server que vous avez appliqué à l'installation de chaque noeud.

Avant de commencer

Important :

- **JM 3.0** WebSphere Application Server traditionnel ne prend actuellement pas en charge Jakarta EE. Voir la déclaration de prise en charge de l'adaptateur de ressources [IBM MQ](#).
- Si vous utilisez WebSphere Application Server 8.5 ou une version antérieure sur une plateforme, n'installez pas l'adaptateur de ressources IBM MQ 8.0 ou version ultérieure dans le serveur d'applications. L'adaptateur de ressources IBM MQ 8.0 ou version ultérieure peut uniquement être déployé dans un serveur d'applications prenant en charge JMS 2.0. Toutefois, WebSphere Application Server 8.5 ou une version antérieure ne prend en charge que JMS 1.1.

Pourquoi et quand exécuter cette tâche

Utilisez cette tâche si l'une des situations suivantes s'applique à votre configuration et que vous souhaitez configurer tous les serveurs de votre environnement pour utiliser la dernière version de l'adaptateur de ressources IBM MQ :

- Les journaux JVM de n'importe quel serveur d'applications de votre environnement affichent les informations de version d'adaptateur de ressources IBM MQ suivantes après l'application de WebSphere Application Server 7.0.0 Fix Pack 1 ou d'une version ultérieure:
`WMSG1703I:RAR version d'implémentation 7.0.0.0-k700-L080820`
- Les journaux JVM de tout serveur d'applications de votre environnement contiennent l'entrée suivante:
`WMSG1625E: Impossible de détecter
le code du fournisseur de messagerie IBM MQ au chemin spécifié < null>`
- Un ou plusieurs noeuds ont été précédemment mis à jour manuellement pour utiliser un niveau de maintenance spécifique de l'adaptateur de ressources IBM MQ qui est désormais remplacé par la version la plus récente de l'adaptateur de ressources contenue dans le niveau de maintenance WebSphere Application Server en cours.

Le répertoire *profile_root* auquel se réfèrent les exemples est le répertoire de base du profil WebSphere Application Server, par exemple `C:\Program Files\IBM\WebSphere\AppServer1`.

Une fois que vous avez effectué les étapes suivantes pour toutes les cellules et les installations à serveur unique de votre environnement, vos serveurs reçoivent automatiquement la maintenance de l'adaptateur de ressources IBM MQ lorsqu'un nouveau groupe de correctifs WebSphere Application Server est appliqué.

Procédure

1. Démarrez le serveur d'applications. Si le profil fait partie d'une configuration de déploiement réseau, démarrez le gestionnaire de déploiement et tous les agents de noeud. Si le profil contient un agent d'administration, démarrez-le.
2. Vérifiez le niveau de maintenance de l'adaptateur de ressources IBM MQ.
 - a) Ouvrez une fenêtre d'invite de commande et accédez au répertoire *profile_root*\bin.
Par exemple, entrez `cd C:\Program Files\IBM\WebSphere\AppServer1\bin`.
 - b) Démarrez l'outil wsadmin en entrant `wsadmin.bat -lang jython`, puis, si vous y êtes invité, entrez votre nom d'utilisateur et votre mot de passe.

c) Entrez la commande suivante, puis appuyez deux fois sur Entrée:

```
wmqInfoMBeansUnsplit = AdminControl.queryNames("WebSphere:type=WMQInfo,*")
wmqInfoMBeansSplit = AdminUtilities.convertToList(wmqInfoMBeansUnsplit)
for wmqInfoMBean in wmqInfoMBeansSplit: print wmqInfoMBean; print AdminControl.invoke(wmqInfoMBean,
'getInfo', '')
```

Vous pouvez également exécuter cette commande dans Jacl. Pour plus d'informations sur la procédure à suivre, voir *Vérification que les serveurs utilisent le niveau de maintenance d'adaptateur de ressources IBM MQ disponible le plus récent* dans la documentation du produit WebSphere Application Server .

d) Recherchez le message WMSG1703I dans la sortie affichée à partir de la commande et vérifiez le niveau de l'adaptateur de ressources.

Par exemple, pour WebSphere Application Server 7.0.1 Fix Pack 5, le message doit être:

```
WMSG1703I: Version d'implémentation RAR 7.0.1.3-k701-103-100812
```

Ce message indique que la version est 7.0.1.3-k701-103-100812, qui correspond au niveau d'adaptateur de ressources correct pour ce groupe de correctifs. Toutefois, si le message suivant s'affiche à la place, cela signifie que vous devez ajuster l'adaptateur de ressources au niveau de maintenance approprié pour WebSphere Application Server 7.0.1 Fix Pack 5.

```
WMSG1703I: Version d'implémentation RAR 7.0.0.0-k700-L080820
```

3. Copiez le script Jython suivant dans un fichier appelé `convertWMQRA.py`, puis enregistrez-le dans le répertoire racine du profil, par exemple `C:\Program Files\IBM\WebSphere\AppServer1\bin`.

```
ras = AdminUtilities.convertToList(AdminConfig.list('J2CResourceAdapter'))

for ra in ras :
    desc = AdminConfig.showAttribute(ra, "description")
    if (desc == "WAS 7.0 Built In MQ Resource Adapter") or (desc == "WAS 7.0.0.1 Built In MQ Resource Adapter"):
        print "Updating archivePath and classpath of " + ra
        AdminConfig.modify(ra, [['archivePath', "${WAS_INSTALL_ROOT}/installedConnectors/wmq.jmsra.rar"]])
        AdminConfig.unsetAttributes(ra, ['classpath'])
        AdminConfig.modify(ra, [['classpath', "${WAS_INSTALL_ROOT}/installedConnectors/wmq.jmsra.rar"]])
        AdminConfig.save()
    #end if
#end for
```

Conseil : Lorsque vous sauvegardez le fichier, assurez-vous qu'il est sauvegardé en tant que fichier python plutôt qu'en tant que fichier texte.

4. Utilisez l'outil `wsadmin` WebSphere Application Server pour exécuter le script Jython que vous venez de créer.

Ouvrez une invite de commande et accédez au répertoire `\bin` dans le répertoire de base du répertoire WebSphere Application Server, par exemple `C:\Program Files\IBM\WebSphere\AppServer1\bin`, puis entrez la commande suivante et appuyez sur Entrée:

```
wsadmin -lang jython -f convertWMQRA.py
```

Si vous y êtes invité, entrez votre nom d'utilisateur et votre mot de passe.

Remarque : Si vous exécutez le script sur un profil qui fait partie d'une configuration de déploiement réseau, le script met à jour tous les profils qui doivent être mis à jour dans cette configuration. Une resynchronisation complète peut être nécessaire si vous avez des incohérences de fichier de configuration préexistantes.

5. Si vous utilisez une configuration de déploiement réseau, vérifiez que les agents de noeud sont entièrement resynchronisés. Pour plus d'informations, voir *Synchronisation des noeuds à l'aide de l'outil de scriptage wsadmin ou Ajout, gestion et suppression de noeuds*.
6. Arrêtez tous les serveurs du profil. Si le profil fait partie d'une configuration de déploiement réseau, arrêtez également tous les membres de cluster de la configuration, arrêtez tous les agents de noeud de la configuration et arrêtez le gestionnaire de déploiement. Si le profil contient un agent d'administration, arrêtez-le.

7. Exécutez la commande **osgiCfgInit** à partir du répertoire *profile_root/bin* .
La commande **osgiCfgInit** réinitialise le cache de classe utilisé par l'environnement d'exécution OSGi. Si le profil fait partie d'une configuration de déploiement réseau, exécutez la commande **osgiCfgInit** à partir du répertoire *profile_root/bin* de chaque profil faisant partie de la configuration.
8. Redémarrez tous les serveurs du profil. Si le profil fait partie d'une configuration de déploiement réseau, redémarrez également tous les membres de cluster de la configuration, redémarrez tous les agents de noeud de la configuration et redémarrez le gestionnaire de déploiement. Si le profil contient un agent d'administration, redémarrez l'agent d'administration.
9. Répétez l'étape 2 pour vérifier que le niveau de l'adaptateur de ressources est désormais correct.

Que faire ensuite

Si vous continuez à rencontrer des problèmes après avoir effectué les étapes décrites dans cette rubrique et que vous avez précédemment utilisé le bouton **Mettre à jour l'adaptateur de ressources** du panneau JMS Paramètres du fournisseur dans la console d'administration WebSphere Application Server pour mettre à jour l'adaptateur de ressources IBM MQ sur tous les noeuds de votre environnement, il est possible que vous rencontriez le problème décrit dans l' [APAR PM10308](#).

Concepts associés

[Utilisation de l'adaptateur de ressources IBM MQ](#)

Informations liées à WebSphere Application Server 8.5.5

[S'assurer que les serveurs utilisent le dernier niveau de maintenance d'adaptateur de ressources IBM MQ disponible](#)

[Synchronisation des noeuds à l'aide de l'outil de scriptage wsadmin](#)

[Ajout, gestion et suppression de noeuds](#)

[JMS Paramètres de fournisseur](#)

Configuration de la propriété JMS PROVIDERVERSION

Le fournisseur de messagerie IBM MQ possède trois modes de fonctionnement : le mode normal, le mode normal avec restrictions et le mode de migration. Vous pouvez définir la propriété JMS **PROVIDERVERSION** pour sélectionner les modes qu'une application JMS utilise pour la publication et l'abonnement.

Pourquoi et quand exécuter cette tâche

La sélection du mode de fonctionnement du fournisseur de messagerie IBM MQ peut être principalement contrôlée en définissant la propriété de fabrique de connexions PROVIDERVERSION. Le mode de fonctionnement peut également être sélectionné automatiquement si aucun mode n'a été spécifié.

La propriété **PROVIDERVERSION** différencie les trois modes de fonctionnement du fournisseur de messagerie IBM MQ :

Le mode normal du fournisseur de messagerie IBM MQ

Le mode normal utilise toutes les fonctions d'un gestionnaire de files d'attente IBM MQ pour implémenter JMS. Ce mode est optimisé pour utiliser l'API JMS 2.0 et ses fonctionnalités.

Le mode normal avec restrictions du fournisseur de messagerie IBM MQ

Le mode normal avec restrictions utilise l'API JMS 2.0 , mais pas les nouvelles fonctions, c'est-à-dire les abonnements partagés, la distribution différée et l'envoi asynchrone.

Le mode de migration du fournisseur de messagerie IBM MQ

Avec le mode de migration, vous pouvez vous connecter à un gestionnaire de files d'attente IBM MQ 8.0 ou version ultérieure, mais aucune des fonctions d'un gestionnaire de files d'attente IBM WebSphere MQ 7.0 ou version ultérieure, telles que la lecture anticipée et la diffusion en flux, n'est utilisée.

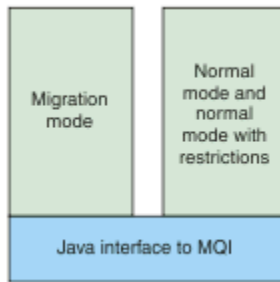


Figure 89. Modes de fournisseur de messagerie

Procédure

Pour configurer la propriété **PROVIDERVERSION** pour une fabrique de connexions spécifique:

- Pour configurer la propriété **PROVIDERVERSION** à l'aide de IBM MQ Explorer, voir [Configuration des gestionnaires de files d'attente et des objets](#).
- Pour configurer la propriété **PROVIDERVERSION** à l'aide de l'outil d'administration JMS, voir [Configuration des gestionnaires de files d'attente et des objets](#).
- Pour configurer la propriété **PROVIDERVERSION** dans une application JMS à l'aide des extensions IBM JMS ou IBM MQ JMS, voir [Création et configuration de fabriques de connexions et de destinations dans une application IBM MQ classes for JMS](#).

Pour remplacer les paramètres de mode du fournisseur de fabrique de connexions pour toutes les fabriques de connexions de la machine virtuelle Java:

- Pour remplacer les paramètres de mode du fournisseur de fabrique de connexions, utilisez la propriété `com.ibm.msg.client.wmq.overrideProviderVersion`

Si vous ne pouvez pas modifier la fabrique de connexions que vous utilisez, vous pouvez utiliser la propriété `com.ibm.msg.client.wmq.overrideProviderVersion` pour remplacer tout paramètre de la fabrique de connexions. Ce remplacement s'applique à toutes les fabriques de connexions de la machine virtuelle Java mais les objets de fabrique de connexions réels ne sont pas modifiés.

Concepts associés

[Identification et résolution des problèmes de version du fournisseur JMS](#)

Référence associée

[PROVIDERVERSION](#)

[Propriétés de fabrique de connexions](#)

[Dépendances entre les propriétés des objets de IBM MQ classes for JMS](#)

Modes de fonctionnement du fournisseur de messagerie IBM MQ

Vous pouvez sélectionner le mode de fonctionnement du fournisseur de messagerie IBM MQ utilisé par une application JMS pour la publication et l'abonnement en définissant la propriété **PROVIDERVERSION** de la fabrique de connexions sur la valeur appropriée. Dans certains cas, la propriété **PROVIDERVERSION** est définie comme non spécifiée, auquel cas le client JMS utilise un algorithme pour déterminer le mode de fonctionnement à utiliser.

PROVIDERVERSION Valeurs de propriété

Vous pouvez définir la propriété **PROVIDERVERSION** de la fabrique de connexions sur l'une des valeurs suivantes:

8 - mode normal

L'application JMS utilise le mode normal. Ce mode utilise toutes les fonctions d'un gestionnaire de files d'attente IBM MQ pour implémenter JMS.

7 - mode normal avec restrictions

L'application JMS utilise le mode normal avec restrictions. Ce mode utilise l'API JMS 2.0, mais pas les nouvelles fonctions telles que les abonnements partagés, la distribution différée ou l'envoi asynchrone.

6 - Mode de migration

L'application JMS utilise le mode de migration. En mode de migration, IBM MQ classes for JMS utilise des fonctions et des algorithmes similaires à ceux qui sont fournis avec IBM WebSphere MQ 6.0.

non spécifié (valeur par défaut)

Le client JMS utilise un algorithme pour identifier le mode de fonctionnement utilisé.

La valeur que vous spécifiez pour la propriété **PROVIDERVERSION** doit être une chaîne. Vous pouvez spécifier l'option 8, 7 ou 6 dans l'un des formats suivants :

- V.R.M.F
- V.R.M
- V.R
- V

où V, R, M et F sont des entiers strictement positifs. Les valeurs R, M et F sont facultatives ; vous pouvez les utiliser si le contrôle à granularité fine est nécessaire. Par exemple, si vous souhaitez utiliser un niveau **PROVIDERVERSION** de 7, vous pouvez définir **PROVIDERVERSION** = 7, 7.0, 7.0.0 ou 7.0.0.0.

Types d'objet de fabrication de connexions

Vous pouvez définir la propriété **PROVIDERVERSION** pour les types d'objet de fabrication de connexions suivants:

- MQConnectionFactory
- Fabrique MQQueueConnection
- Fabrique MQTopicConnection
- MQXAConnectionFactory
- Fabrique MQXAQueueConnection
- Fabrique MQXAQueueConnection
- Fabrique MQXAQueueConnection
- Fabrique MQXATopicConnection

Pour plus d'informations sur ces différents types de fabrication de connexions, voir [«Configuration des objets JMS et Jakarta Messaging à l'aide des outils d'administration»](#), à la page 753.

Concepts associés

[Fournisseur de messagerie IBM MQ](#)

PROVIDERVERSION mode normal

Le mode normal utilise toutes les fonctions d'un gestionnaire de files d'attente IBM MQ pour implémenter JMS. Ce mode est optimisé pour utiliser l'API JMS 2.0 et ses fonctionnalités.

La diagramme ci-dessous présente les vérifications effectuées par le client JMS pour déterminer si une connexion en mode normal peut être créée.

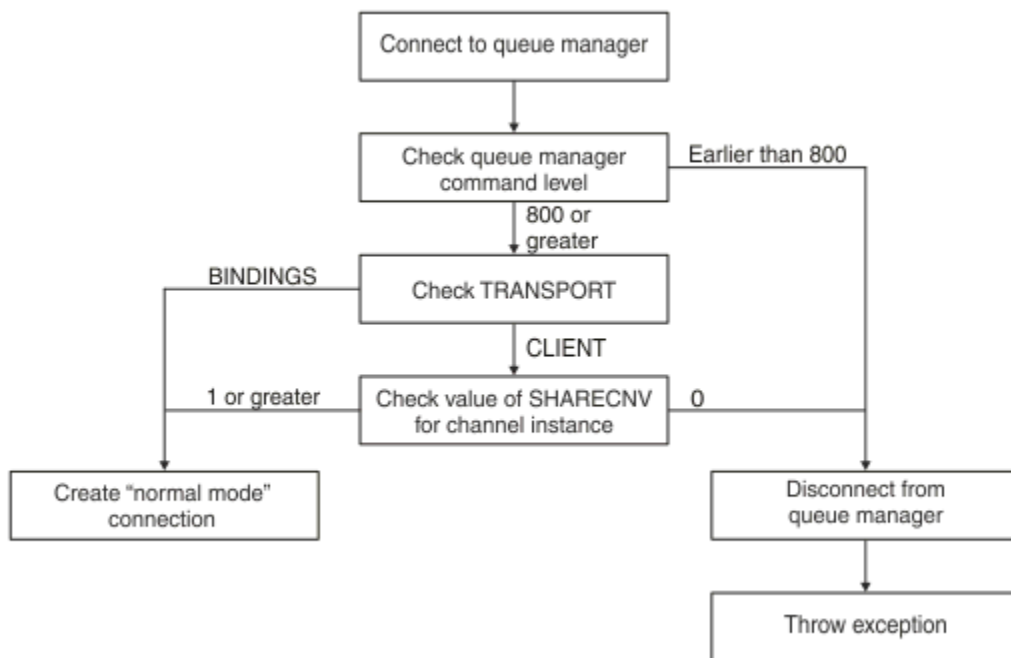


Figure 90. PROVIDERVERSION mode normal

Si le gestionnaire de files d'attente indiqué dans les paramètres de la fabrique de connexions a un niveau de commande 800 ou supérieur, et si la propriété **TRANSPORT** de la fabrique de connexions est définie sur BINDINGS, une connexion en mode normal est créée sans autre vérification de propriétés.

Si le gestionnaire de files d'attente indiqué dans les paramètres de la fabrique de connexions a un niveau de commande 800 ou supérieur et si la propriété **TRANSPORT** est définie sur CLIENT, la propriété **SHARECNV** sur le canal de connexion serveur est également vérifiée. Cette vérification est nécessaire car le fournisseur de messagerie IBM MQ en mode normal utilise la fonction de partage de conversations. Ainsi, pour qu'une tentative de connexion en mode normal aboutisse, la propriété **SHARECNV** qui contrôle le nombre de conversations pouvant être partagées doit avoir la valeur 1 ou supérieure.

Si toutes les vérifications affichées dans le diagramme aboutissent, une connexion en mode normal au gestionnaire de files d'attente est créée et toutes les API et fonctions JMS 2.0, c'est-à-dire l'envoi asynchrone, la distribution différée et l'abonnement partagé, peuvent ensuite être utilisées.

Une tentative de création d'une connexion en mode normal échoue pour l'une des raisons suivantes :

- Le gestionnaire de files d'attente indiqué dans les paramètres de la fabrique de connexions a un niveau de commande inférieur à 800. Dans ce cas, la méthode `createConnection` échoue avec l'exception `JMSFMQ0003`.
- La propriété **SHARECNV** sur le canal de connexion serveur est définie sur 0. Si cette propriété n'a pas une valeur supérieure ou égale à 1, la méthode `createConnection` échoue avec une exception `JMSCC5007`.

Référence associée

[Dépendances entre les propriétés des objets IBM MQ classes for JMS](#)

[DEFINE CHANNEL \(propriété SHARECNV\)](#)

[TRANSPORT](#)

Mode normal PROVIDERVERSION avec restrictions

Le mode normal avec restrictions utilise l'API JMS 2.0, mais pas les nouvelles fonctions IBM MQ 8.0 ou ultérieures, telles que les abonnements partagés, la distribution différée ou l'envoi asynchrone.

Le diagramme suivant montre les vérifications effectuées par le client JMS pour déterminer si un mode normal avec des restrictions de connexion peut être créé.

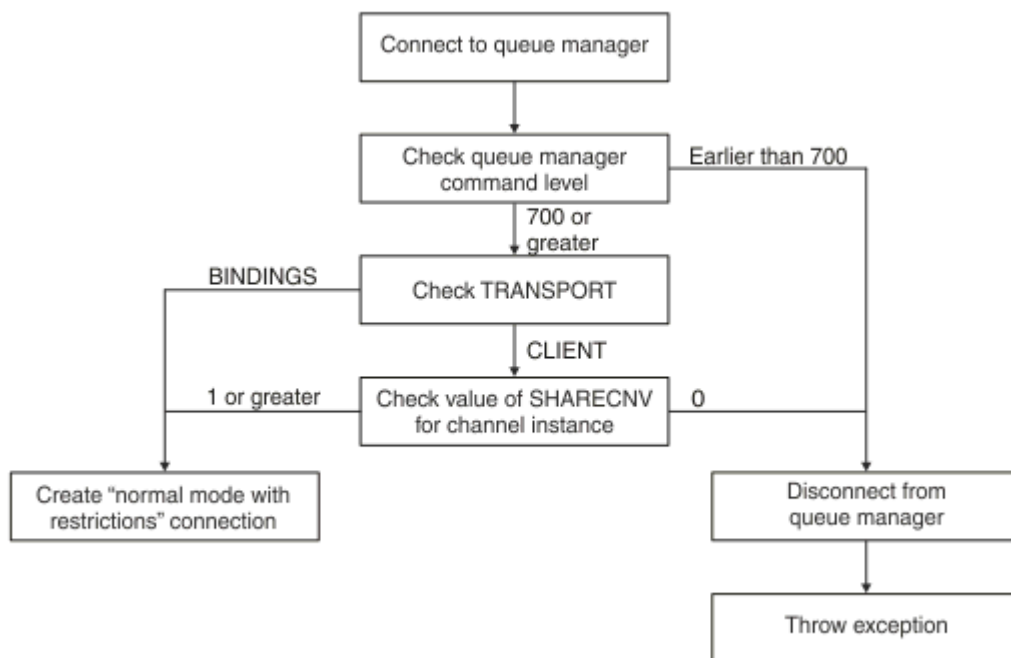


Figure 91. PROVIDERVERSION mode normal avec restrictions

Si le gestionnaire de files d'attente spécifié dans les paramètres de la fabrique de connexions possède un niveau de commande supérieur ou égal à 700 et que la propriété **TRANSPORT** de la fabrique de connexions est définie sur BINDINGS, une connexion en mode normal est créée sans vérifier d'autres propriétés.

Si le gestionnaire de files d'attente spécifié dans les paramètres de fabrique de connexions possède un niveau de commande 700 ou supérieur et que la propriété **TRANSPORT** est définie sur CLIENT, la propriété **SHARECNV** sur le canal de connexion serveur est également vérifiée. Cette vérification est nécessaire car le mode normal du fournisseur de messagerie IBM MQ avec restrictions utilise la fonction de partage des conversations. Par conséquent, pour qu'une tentative de connexion en mode normal avec restrictions aboutisse, la propriété **SHARECNV**, qui contrôle le nombre de conversations pouvant être partagées, doit avoir une valeur supérieure ou égale à 1.

Si toutes les vérifications affichées dans le diagramme aboutissent, un mode normal avec des restrictions de connexion au gestionnaire de files d'attente est créé et vous pouvez alors utiliser l'API JMS 2.0, mais pas les fonctions d'envoi asynchrone, de distribution différée ou d'abonnement partagé.

Une tentative de création d'un mode normal avec des restrictions de connexion échoue pour l'une des raisons suivantes:

- Le niveau de commande du gestionnaire de files d'attente spécifié dans les paramètres de fabrique de connexions est antérieur à 700. Dans ce cas, la méthode `createConnection` échoue avec l'exception JMSFCC5008.
- La propriété **SHARECNV** sur le canal de connexion serveur est définie sur 0. Si cette propriété n'a pas une valeur supérieure ou égale à 1, la méthode `createConnection` échoue avec une exception JMSSC5007.

Référence associée

Dépendances entre les propriétés des objets IBM MQ classes for JMS

[DEFINE CHANNEL \(propriété SHARECNV\)](#)

[TRANSPORT](#)

Mode de migration PROVIDERVERSION

Pour le mode de migration, IBM MQ classes for JMS utilise des fonctions et des algorithmes similaires à ceux qui sont fournis avec IBM WebSphere MQ 6.0, tels que la publication / l'abonnement en file

d'attente, la sélection implémentée côté client, les canaux non multiplex et l'interrogation utilisée pour implémenter les programmes d'écoute.

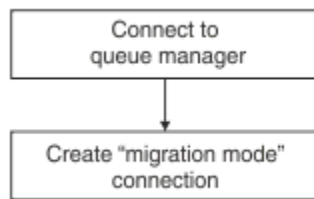


Figure 92. Mode de migration PROVIDERVERSION

Si vous souhaitez vous connecter à WebSphere Message Broker 6.0 ou WebSphere Message Broker 6.1 à l'aide de IBM MQ Enterprise Transport version 6.0, vous devez utiliser le mode de migration.

Vous pouvez vous connecter à un gestionnaire de files d'attente IBM MQ 8.0 à l'aide du mode de migration, mais aucune des nouvelles fonctions d'un gestionnaire de files d'attente IBM MQ classes for JMS n'est utilisée, par exemple, la lecture anticipée ou la diffusion en flux. Si un client IBM MQ 8.0 ou version ultérieure se connecte à un gestionnaire de files d'attente IBM MQ 8.0 ou version ultérieure sur une plateforme répartie, **z/OS** ou un gestionnaire de files d'attente IBM MQ for z/OS 8.0 ou version ultérieure, la sélection des messages est effectuée par le gestionnaire de files d'attente plutôt que par le système client.

Si le mode de migration du fournisseur de messagerie IBM MQ est spécifié et que IBM MQ classes for JMS tente d'utiliser l'une des API JMS 2.0 , l'appel de méthode d'API échoue avec l'exception JM5CC5007.

Référence associée

Dépendances entre les propriétés des objets IBM MQ classes for JMS TRANSPORT

PROVIDERVERSION non spécifié

Lorsque la propriété **PROVIDERVERSION** d'une fabrique de connexions n'est pas spécifiée, le client JMS utilise un algorithme pour déterminer quel mode de fonctionnement est utilisé pour la connexion au gestionnaire de files d'attente. Une fabrique de connexions créée dans l'espace de nom JNDI avec une version précédente de IBM MQ classes for JMS prend la valeur non spécifiée lorsque la fabrique de connexions est utilisée avec la nouvelle version de IBM MQ classes for JMS.

Si la propriété **PROVIDERVERSION** n'est pas spécifiée, l'algorithme est utilisé lorsque la méthode `createConnection` est appelée. L'algorithme vérifie un certain nombre de propriétés de fabrique de connexions pour déterminer si le mode normal du fournisseur de messagerie IBM MQ , le mode normal avec restrictions ou le mode de migration du fournisseur de messagerie IBM MQ est requis. Le mode normal est toujours tenté en premier, puis le mode normal avec des restrictions. Si aucun de ces types de connexion ne peut être établi, le client JMS se déconnecte du gestionnaire de files d'attente, puis se reconnecte au gestionnaire de files d'attente pour tenter une connexion en mode de migration.

Vérification des propriétés BROKERVER, BROKERQMGR, PSMODE et BROKERCONQ

La vérification des valeurs de propriété commence par la propriété **BROKERVER** , comme illustré dans la Figure 1.

Si la propriété **BROKERVER** est définie sur V1, la propriété **TRANSPORT** est cochée ensuite, comme illustré dans la Figure 2. Toutefois, si la propriété **BROKERVER** est définie sur V2, la vérification supplémentaire illustrée dans la Figure 1 est effectuée avant la vérification de la propriété **TRANSPORT** .

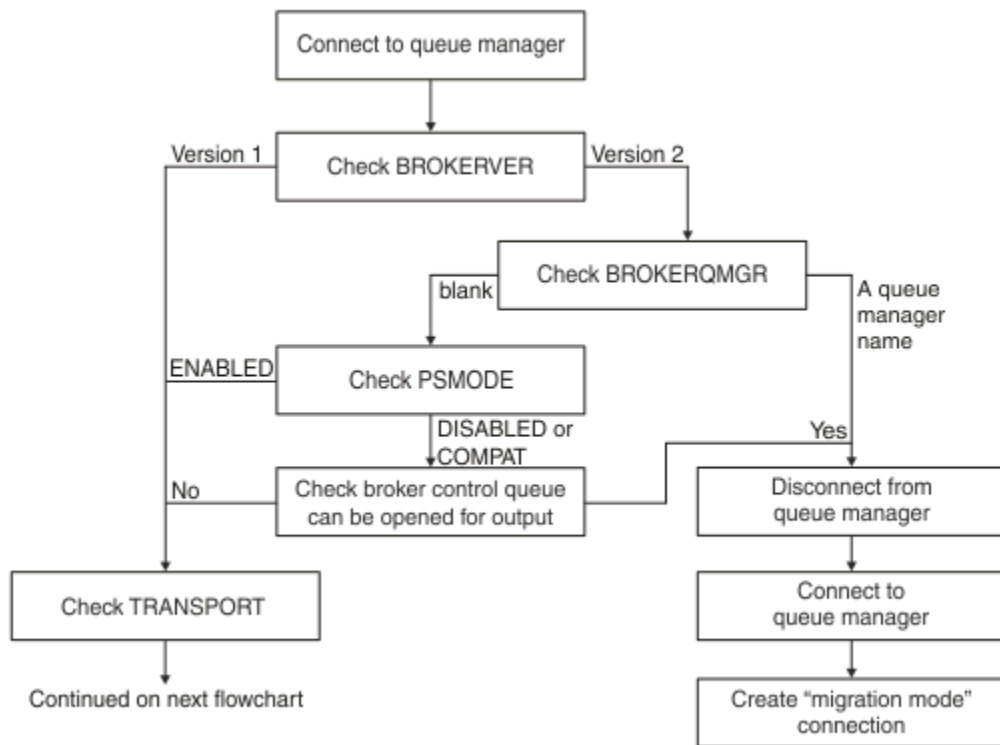


Figure 93. PROVIDERVERSION non spécifiée

Si la propriété **BROKERVER** est définie sur V2, pour qu'une connexion en mode normal soit possible, la propriété **BROKERQMGR** doit être vide. De plus, l'attribut **PSMODE** du gestionnaire de files d'attente doit être défini sur **ENABLED** ou la file d'attente de contrôle du courtier spécifiée par la propriété **BROKERCONQ** ne doit pas pouvoir être ouverte pour la sortie.

Si les valeurs de propriété sont définies comme requises pour une connexion en mode normal, la vérification passe ensuite à la propriété **TRANSPORT**, comme illustré dans la [Figure 2](#).

Si les valeurs de propriété ne sont pas définies comme requises pour une connexion en mode normal, le client JMS se déconnecte du gestionnaire de files d'attente, puis se reconnecte et crée une connexion en mode de migration. Cela se produit dans les cas suivants:

- Si la propriété **BROKERQMGR** est vide et que l'attribut **PSMODE** du gestionnaire de files d'attente est défini sur **COMPAT** ou **DISABLED** et que la file d'attente de contrôle du courtier spécifiée par la propriété **BROKERCONQ** peut être ouverte pour la sortie (c'est-à-dire **MQOPEN** pour que la sortie aboutisse).
- Si la propriété **BROKERQMGR** indique un nom de file d'attente.

Vérification de la propriété **TRANSPORT** et du niveau de commande

La [Figure 2](#) présente les vérifications effectuées pour la propriété **TRANSPORT** et le niveau de commande du gestionnaire de files d'attente.

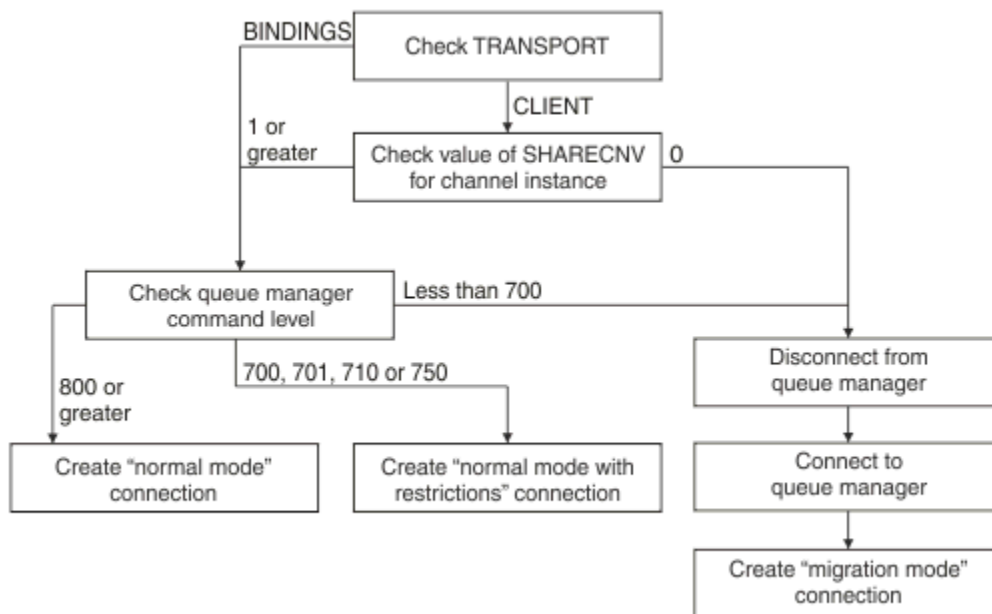


Figure 94. PROVIDERVERSION non spécifié (suite)

Une connexion en mode normal est créée dans l'un des cas suivants:

- La propriété **TRANSPORT** de la fabrique de connexions est définie sur BINDINGS, et le niveau de commande du gestionnaire de files d'attente est 800 ou supérieur.
- La propriété **TRANSPORT** est définie sur CLIENT, la propriété **SHARECNV** sur le canal de connexion serveur a une valeur supérieure ou égale à 1 et le gestionnaire de files d'attente a un niveau de commande supérieur ou égal à 800.

Si le niveau de commande du gestionnaire de files d'attente est 750, un mode normal avec des restrictions de connexion au gestionnaire de files d'attente est créé.

Une connexion en mode migration est également créée si la propriété **TRANSPORT** est définie sur CLIENT et que la propriété **SHARECNV** sur le canal de connexion serveur a la valeur 0.

Référence associée

Dépendances entre les propriétés des objets IBM MQ classes for JMS

[ALTER QMGR \(attribut PSMODE\)](#)

[BROKERCONQ](#)

[BROKERQMGR](#)

[BROKERVER](#)

[DEFINE CHANNEL \(propriété SHARECNV\)](#)

[TRANSPORT](#)

Configuration des informations de version du fournisseur dans WebSphere Application Server

Pour configurer les informations de version du fournisseur dans WebSphere Application Server, vous pouvez utiliser la console d'administration ou les commandes wsadmin.

Procédure

Pour configurer les informations de version du fournisseur pour un objet de spécification d'activation ou de fabrique de connexions IBM MQ dans WebSphere Application Server, voir *Informations associées* pour obtenir des liens vers d'autres informations dans la documentation du produit WebSphere Application Server .

Informations liées à WebSphere Application Server 8.5.5

Paramètres de la fabrique de connexions du fournisseur de messagerie IBM MQ
commande **`createWMQConnectionFactory`**

Paramètres de spécification d'activation du fournisseur de messagerie IBM MQ
commande **`createWMQActivationSpec`**

Informations liées à WebSphere Application Server 8.0.0

Paramètres de la fabrique de connexions du fournisseur de messagerie IBM MQ
commande **`createWMQConnectionFactory`**

Paramètres de spécification d'activation IBM MQ
commande **`createWMQActivationSpec`**

Informations liées à WebSphere Application Server 7.0.0

Paramètres de la fabrique de connexions du fournisseur de messagerie IBM MQ
commande **`createWMQConnectionFactory`**

Paramètres de spécification d'activation IBM MQ
commande **`createWMQActivationSpec`**

Suppression d'abonnements durables WebSphere Application Server

Si vous utilisez le fournisseur de messagerie IBM MQ avec WebSphere Application Server 7.0 et la WebSphere Application Server 8.0, les abonnements durables qui sont créés par des applications de bean géré par message liées à des spécifications d'activation ne sont pas supprimés. Les abonnements durables peuvent être supprimés à l'aide de l'utilitaire de ligne de commande IBM MQ Explorer ou IBM MQ .

A propos de cette tâche

Une application de bean géré par message qui supprime un abonnement durable peut être configurée pour utiliser un port d'écoute ou une spécification d'activation, à condition que l'application s'exécute dans une instance WebSphere Application Server 7.0 ou WebSphere Application Server 8.0 qui utilise le mode normal du fournisseur de messagerie [IBM MQ](#) pour se connecter à IBM MQ.

Si l'application de bean géré par message est liée à un port d'écoute, le fournisseur de messagerie IBM MQ crée l'abonnement durable pour l'application lorsque l'application est démarrée pour la première fois. L'abonnement durable est supprimé lorsque l'application de bean géré par message est désinstallée d'un serveur d'applications, et le serveur d'applications est redémarré.

Le fonctionnement d'une application de bean géré par message liée à une spécification d'activation est légèrement différent. L'abonnement durable est créé pour l'application la première fois que cette dernière est démarrée. Toutefois, il n'est pas supprimé lorsque l'application est désinstallée et que le serveur d'applications est redémarré.

Cela peut entraîner la persistance d'un certain nombre d'abonnements durables sur un moteur de publication / abonnement IBM MQ pour les applications qui ne sont plus installées dans un système WebSphere Application Server . Ces abonnements sont appelés "abonnements orphelins" et peuvent être sources d'erreurs dans le gestionnaire de files d'attente lors de l'exécution du moteur de publication/abonnement.

Lorsqu'un message est publié dans une rubrique, le moteur de publication/abonnement IBM MQ génère une copie de ce message pour chaque abonnement durable qui est enregistré sur cette rubrique et le place dans une file d'attente interne. Les applications qui utilisent cet abonnement durable sélectionnent et consomment le message provenant de cette file d'attente interne.

Si l'application de bean géré par message qui utilisait cet abonnement durable n'est plus installée, les copies des messages publiés pour l'application continueront à être générées. Cependant, ces messages ne seront jamais traités, ce qui signifie qu'un grand nombre de messages restant dans la file d'attente interne ne seront jamais supprimés.

Avant de commencer

Les abonnements qui sont enregistrés avec le moteur de publication/abonnement IBM MQ seront associés à un nom d'abonnement.

Les abonnements durables créés par le fournisseur de messagerie WebSphere Application Server d'IBM MQ pour les beans gérés par message qui sont liés à des spécifications d'activation auront un nom d'abonnement au format suivant :

```
JMS:queue manager name:client identifier:subscription name
```

où :

nom du gestionnaire de files d'attente

Nom du gestionnaire de files d'attente IBM MQ dans lequel s'exécute le moteur de publication/abonnement.

identificateur de client

Valeur de la propriété Client ID de la spécification d'activation à laquelle est liée le bean géré par message.

nom d'abonnement

Valeur de la propriété Subscription name pour la spécification d'activation devant être utilisée par l'application de bean géré par message.

Par exemple, supposons qu'une spécification d'activation a été configurée pour se connecter au gestionnaire de files d'attente QMtest. Les propriétés définies pour la spécification d'activation sont les suivantes :

- Client ID = IDClientTest
- Subscription name = abonnementDurable1

Si un bean géré par message qui prend un abonnement durable est lié à cette spécification d'activation, un abonnement est créé sur le moteur de publication/abonnement IBM MQ dans le gestionnaire de files d'attente QMtest ayant le nom d'abonnement suivant :

- JMS:testQM:testClientID:durableSubscription1

Les abonnements qui ont été enregistrés avec le moteur de publication/abonnement IBM MQ pour un gestionnaire de files d'attente donné peuvent être visualisés de deux manières :

- La première option consiste à utiliser MQ Explorer. Lorsque MQ Explorer a été connecté à un gestionnaire de files d'attente utilisé pour le travail de publication / abonnement, la liste des abonnés actuellement enregistrés avec le moteur de publication / abonnement peut être affichée en cliquant sur l'entrée IBM WebSphere MQ ->queue manager name-> Subscriptions dans le panneau de navigation.
- L'autre moyen d'afficher les abonnements qui ont été enregistrés avec un moteur de publication / abonnement consiste à utiliser l' IBM MQ utilitaire de ligne de commande **runmqsc** et à exécuter la commande **display sub**. Pour ce faire, ouvrez une invite de commande, accédez au répertoire *WebSphere MQ\bin* et entrez la commande suivante pour démarrer **runmqsc**:

```
- runmqsc queue manager name
```

Lorsque l'utilitaire **runmqsc** a démarré, entrez la commande suivante pour afficher la liste de tous les abonnements durables enregistrés avec le moteur de publication/abonnement exécuté dans le gestionnaire de files d'attente auquel **runmqsc** est connecté :

```
- display sub(*) durable
```

Pour vérifier si les abonnements durables enregistrés avec les moteurs de publication/abonnement sont encore actifs :

1. Générez la liste des abonnements durables qui ont été enregistrés avec le moteur de publication/abonnement.
2. Pour chaque abonnement durable :

- recherchez le nom d'abonnement pour l'abonné durable et notez les valeurs *identificateur de client* et *nom d'abonnement*.
- Recherchez les systèmes WebSphere Application Server qui se connectent à ce moteur de publication/abonnement. Déterminez s'il existe des spécifications d'activation définies dont la propriété Client ID correspond à la valeur *identificateur de client* et la propriété Subscription name correspond à la valeur *nom d'abonnement*.
- S'il n'existe aucune spécification d'activation dont les propriétés Client ID et Subscription name correspondent aux zones *identificateur de client* et *nom d'abonnement* dans le nom d'abonnement IBM MQ, aucune spécification d'activation n'utilise cet abonnement durable. L'abonnement durable peut être supprimé.
- Si une spécification d'activation correspondant au nom d'abonnement durable est définie, la vérification finale à effectuer consiste à vérifier si une application de bean géré par message utilise cette spécification d'activation. Pour cela, procédez comme suit :
 - Prenez note du nom JNDI de la spécification d'activation qui a pris l'abonnement durable que vous vérifiez.
 - Affichez le panneau de configuration dans la console d'administration WebSphere Application Server pour chaque application de bean géré par message qui est installée.
 - Cliquez sur le lien des liaisons de programme d'écoute du bean géré par message dans le panneau de configuration.
 - Un tableau contenant des informations sur l'application de bean géré par message s'affiche. Si le bouton d'option de spécification d'activation est sélectionné dans la colonne Liaisons et si la zone Nom JNDI de la ressource cible contient le nom JNDI pour la spécification d'activation qui a pris l'abonnement durable, l'abonnement est encore utilisé et il ne peut pas être supprimé.
 - Si aucune application de bean géré par message utilisant la spécification d'activation n'est trouvée, l'abonnement durable peut être supprimé.

Procédure

Une fois qu'un abonnement durable "orphelin" a été identifié, il peut être supprimé à l'aide de IBM MQ Explorer ou de l'utilitaire de ligne de commande IBM MQ **runmqsc**.

Pour supprimer un abonnement durable "orphelin" à l'aide de IBM MQ Explorer :

1. Mettez en évidence l'entrée pour l'abonnement
2. Cliquez avec le bouton droit de la souris sur l'entrée et sélectionnez **Supprimer ...** dans le menu. Une fenêtre de confirmation apparaît.
3. Vérifiez si le nom d'abonnement affiché dans la fenêtre de confirmation est correct et cliquez sur **Oui**.

IBM MQ Explorer supprime alors l'abonnement du moteur de publication/abonnement et nettoie les ressources internes associées à celui-ci (comme les messages non traités qui ont été publiés pour la rubrique sur laquelle l'abonnement durable a été enregistré).

Pour supprimer un abonnement durable "orphelin" à l'aide de l' IBM MQ utilitaire de ligne de commande **runmqsc**, la commande **delete sub** doit être exécutée:

1. Ouvrez une session d'invite de commande
2. Accédez au répertoire *IBM MQ\bin*
3. Entrez la commande suivante pour démarrer **runmqsc** :

```
runmqsc queue manager name
```

4. Lorsque l'utilitaire **runmqsc** a démarré, entrez :

```
delete sub(Subscription name)
```

où *nom d'abonnement* est le nom d'abonnement de l'abonnement durable au format suivant :

- `JMS:queue manager name:client identifier:subscription name`

Configuration de Managed File Transfer

Vous pouvez configurer les fonctions d' Managed File Transfer après l'installation.

Vous pouvez tirer parti des solutions à haute disponibilité IBM MQ pour améliorer la résilience de votre configuration Managed File Transfer . Si vos agents utilisent des gestionnaires de files d'attente de données répliquées (RDQM), vous devez les configurer pour qu'ils utilisent la fonction d'adresse IP flottante. Cela signifie que les agents utilisent la même adresse IP pour communiquer avec l'une des trois instances de gestionnaire de files d'attente de données répliquées en cours d'exécution et se reconnectent automatiquement lors de la reprise en ligne (voir [Haute disponibilité des gestionnaires de files d'attente de données répliquées](#) et [Création et suppression d'une adresse IP flottante](#)). Si vous utilisez la solution de gestionnaire de files d'attente multi-instance, les applications utilisent une adresse IP différente pour communiquer avec chaque instance, qui est gérée par la reconnexion du client lors de la reprise en ligne (voir «[Gestionnaires de files d'attente multi-instance](#)», à la page 547 et «[Reconnexion canal et client](#)», à la page 608).

Concepts associés

[Conseils et astuces pour l'utilisation d' Managed File Transfer](#)

Tâches associées

[Surveillance des ressources MFT](#)

[Personnalisation de MFT avec des exits utilisateur](#)

[Configuration de MQMFTCredentials.xml](#)

[Sécurisation de Managed File Transfer](#)

[Spécification des programmes à exécuter avec MFT](#)

[Dépannage Managed File Transfer](#)

[Administration de Managed File Transfer](#)

Référence associée

[Commandes MFT](#)

[Le fichier MFTagent.properties](#)

[Récupération et redémarrage de MFT](#)

Options de configuration de MFT sur Multiplatforms

Managed File Transfer fournit un ensemble de fichiers de propriétés qui contiennent des informations clés sur votre configuration et qui sont nécessaires au fonctionnement. Ces fichiers de propriétés se trouvent dans le répertoire de configuration que vous avez défini lors de l'installation du produit.

Vous pouvez disposer de plusieurs ensembles d'options de configuration, chaque ensemble d'options de configuration contenant un ensemble de répertoires et de fichiers de propriétés. Les valeurs définies dans ces fichiers de propriétés sont utilisées comme paramètres par défaut pour toutes les commandes Managed File Transfer , sauf si vous spécifiez explicitement une valeur différente sur la ligne de commande.

Pour modifier l'ensemble d'options de configuration par défaut que vous utilisez, vous pouvez utiliser la commande **fteChangeDefaultConfigurationOptions** . Pour modifier l'ensemble des options de configuration que vous utilisez pour une commande individuelle, vous pouvez utiliser le paramètre **-p** avec n'importe quelle commande Managed File Transfer .

Le nom d'un ensemble d'options de configuration est le nom du gestionnaire de file d'attente de coordination et il est recommandé de ne pas le modifier. Toutefois, il est possible de modifier le nom d'un ensemble d'options de configuration, mais vous devez modifier le nom des répertoires `config` et `logs` . Dans les exemples suivants, le nom de l'ensemble d'options de configuration est représenté par `coordination_qmgr_name`.

Structure de répertoire des options de configuration

Lorsque vous configurez le produit, les répertoires et les fichiers de propriétés sont créés dans la structure suivante du répertoire de configuration. Vous pouvez également modifier ces répertoires et fichiers de propriétés à l'aide des commandes suivantes: **fteSetupCoordination**, **fteSetupCommands**, **fteChangeDefaultConfiguration** et **fteCreateAgent**.

```
MQ_DATA_PATH/mqft/  
  config/  
    coordination_qmgr_name/  
      coordination.properties  
      command.properties  
    agents/  
      agent_name/  
        agent.properties  
        exits  
    loggers/  
      logger_name  
        logger.properties  
  installations/  
    installation_name/  
      installation.properties
```

Le répertoire *coordination_qmgr_name* est un répertoire d'options de configuration. Il peut y avoir plusieurs répertoires d'options de configuration dans le répertoire de configuration. Le répertoire *nom_agent* est un répertoire d'agent. Outre le fichier *agent.properties*, ce répertoire contient le répertoire *exits*, qui est l'emplacement par défaut des routines d'exit utilisateur et des différents fichiers XML générés par les commandes **fteCreateBridgeAgent** et **fteCreateCDAgent**. Il peut y avoir plusieurs répertoires d'agent dans le répertoire *agents* d'un ensemble d'options de configuration.

fichiers de propriétés

installation.properties

Le fichier *installation.properties* indique le nom de votre ensemble d'options de configuration par défaut. Cette entrée renvoie Managed File Transfer vers un ensemble structuré de répertoires et de fichier de propriétés qui contiennent la configuration à utiliser. Généralement, le nom d'un ensemble d'options de configuration est le nom du gestionnaire de files d'attente de coordination associé. Pour plus d'informations sur le fichier *installation.properties*, voir [The MFT installation.properties file](#).

coordination.properties

Le fichier *coordination.properties* indique les détails de connexion au gestionnaire de file d'attente de coordination. Etant donné que plusieurs installations Managed File Transfer peuvent partager le même gestionnaire de files d'attente de coordination, vous pouvez utiliser un lien symbolique vers un fichier *coordination.properties* commun sur une unité partagée. Pour plus d'informations sur le fichier *coordination.properties*, voir [Le fichier MFT coordination.properties](#).

command.properties

Le fichier MFT *command.properties* spécifie le gestionnaire de files d'attente de commandes auquel vous devez vous connecter lorsque vous émettez des commandes et les informations dont Managed File Transfer a besoin pour contacter ce gestionnaire de files d'attente. Pour plus d'informations sur le fichier *command.properties*, voir [Le fichier MFT command.properties](#).

agent.properties

Chaque Managed File Transfer Agent possède son propre fichier de propriétés, *agent.properties*, qui doit contenir les informations qu'un agent utilise pour se connecter à son gestionnaire de files d'attente. Le fichier *agent.properties* peut également contenir des propriétés qui modifient le comportement de l'agent. Pour plus d'informations sur le fichier *agent.properties*, voir [Le fichier MFT agent.properties](#).

logger.properties

Le fichier `logger.properties` spécifie les propriétés de configuration des consigneurs. Pour plus d'informations sur le fichier `logger.properties`, voir [Propriétés de configuration du consigneurMFT](#).

Fichiers de propriétés et pages de codes

Le contenu de tous les fichiers de propriétés Managed File Transfer doit rester en anglais (Etats-Unis) en raison d'une limitation de Java. Si vous éditez des fichiers de propriétés sur un système autre que l'anglais américain, vous devez utiliser des séquences d'échappement Unicode.

Référence associée

[Propriétés SSL/TLS pour MFT](#)

[Propriétés système Java pour MFT](#)

[fteChangeDefaultConfigurationOptions](#)

[Commandes fteSetup: créez le fichier MFT `command.properties`](#)

[fteSetupCoordination](#)

[fteCreateAgent](#)

▶ z/OS

MFT configuration options on z/OS

The Managed File Transfer configuration options on z/OS are the same as the options for distributed platforms.

For more information about configuration options on [Multiplateformes](#), see [“Options de configuration de MFT sur Multiplatforms”](#) on page 787.

On z/OS, the configuration location is defined by the environment variable `BFG_DATA`. If a configuration does not already exist under the z/OS UNIX System Services directory that is referenced by `BFG_DATA`, the `BFGCUSTOM` JCL script of an MFT command PDSE library data set generates the jobs required to create the configuration. The configuration is then created when you run these generated jobs. Configuration creation relies on `BFG_DATA` referencing an existing directory that is accessible.

You can also create and maintain a configuration by using the same **fte** commands that are available on both Multiplatforms and z/OS. For a list of the **fte** commands, see [MFT commands](#).

Related concepts

[“Options de configuration de MFT sur Multiplatforms”](#) on page 787

Managed File Transfer fournit un ensemble de fichiers de propriétés qui contiennent des informations clés sur votre configuration et qui sont nécessaires au fonctionnement. Ces fichiers de propriétés se trouvent dans le répertoire de configuration que vous avez défini lors de l'installation du produit.

[“Creating an agent”](#) on page 806

You need to copy the PDSE to make the agent-specific PDSE, for example `user.MFT.AGENT1`. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

[“Defining the coordination queue manager”](#) on page 804

Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

Related tasks

▶ z/OS

[Configuring `MQMFTCredentials.xml` on z/OS](#)

[“Updating an existing MFT Agent or Logger command data set on z/OS”](#) on page 807

You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

Managed File Transfer components

Redistributable Managed File Transfer package fournit le Redistributable Managed File Transfer Agent, que vous pouvez configurer pour vous connecter à une infrastructure IBM MQ existante et permettre aux utilisateurs de transférer des fichiers sans avoir à installer IBM MQ. Depuis la IBM MQ 9.3.0, le package redistribuable inclut également Redistributable Managed File Transfer Logger.

Avant de commencer

Pour plus d'informations sur les termes de la licence redistribuable pour les Redistributable Managed File Transfer Agent et les Redistributable Managed File Transfer Logger, voir [IBM MQ Redistributable Components](#).

Les composants Redistributable Managed File Transfer package fournissent les fonctionnalités de Managed File Transfer avec les exceptions suivantes:

- Pour le Redistributable Managed File Transfer Agent, la connexion en mode liaison aux gestionnaires de files d'attente de coordination, de commande et d'agent n'est pas prise en charge, vous devez utiliser la connexion en mode client. Lorsque vous émettez des commandes, vous devez fournir les paramètres facultatifs lorsque vous utilisez le Managed File Transfer installé dans le cadre de IBM MQ: hôte du gestionnaire de files d'attente, port, nom et nom du canal.
- Redistributable Managed File Transfer Logger prend en charge les consignateurs de type FILE uniquement, en se connectant en mode client au gestionnaire de files d'attente de coordination uniquement. La connexion en mode client au gestionnaire de files d'attente de coordination pour un consignateur de base de données n'est pas prise en charge. Si vous avez besoin d'une connexion en mode liaisons, vous devez utiliser une installation standard de IBM MQ.
- Depuis la IBM MQ 9.3.0, la commande **fteCreateCDAgent.cmd** n'est pas incluse. Pour obtenir la liste complète des commandes disponibles, voir [Installed MFT command sets](#).
- Managed File TransferConnect:Direct n'est pas pris en charge.
- IBM MQ Explorer n'est pas inclus.

Windows

Vous devez installer les bibliothèques Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019, disponibles à partir de Microsoft, sur votre système pour utiliser le Redistributable Managed File Transfer Agent. Voir [Les derniers téléchargements Visual C++ pris en charge](#).

Depuis la IBM MQ 9.3.0, les bibliothèques Microsoft Visual C++ Redistributable for Visual Studio 2015, 2017 and 2019 sont également requises pour Redistributable Managed File Transfer Logger.

Remarque : Advanced Message Security n'est pas pris en charge avec Redistributable Managed File Transfer package.

Pourquoi et quand exécuter cette tâche

Vous pouvez éventuellement télécharger le fichier Redistributable Managed File Transfer package et configurer Redistributable Managed File Transfer Agent pour qu'il se connecte à une infrastructure IBM MQ existante afin de permettre aux utilisateurs de transférer des fichiers entre leur environnement local et l'infrastructure IBM MQ existante sans qu'ils aient besoin d'installer IBM MQ pour obtenir la fonctionnalité Managed File Transfer .

Depuis la IBM MQ 9.3.0, Redistributable Managed File Transfer package inclut également Redistributable Managed File Transfer Logger, qui vous permet de configurer un consignateur de fichiers pour la connexion en mode client au gestionnaire de files d'attente de coordination.

Procédure

1. Téléchargez le [package d'agent IBM MQ redistribuable Managed File Transfer](#) à partir de Fix Central.
 - a) Choisissez le package correspondant à votre système d'exploitation.

Les noms des fichiers archive ou .zip décrivent leur contenu et leur niveau de maintenance. Les noms de fichier sont au format suivant:

- **Windows** V.R.M.F-IBM-MQFA-Redist-Win64
- **Linux** V.R.M.F-IBM-MQFA-Redist-LinuxX64
- **Linux** V.R.M.F-IBM-MQFA-Redist-LinuxS390X
- **Linux** V.R.M.F-IBM-MQFA-Redist-LinuxPPC64LE

où V.R.M.F est le numéro de version, par exemple 9.2.0.0 ou 9.2.1.0.

b) Identifiez le répertoire dans lequel vous souhaitez extraire le package, par exemple:

- **Windows** C:\MFTZ
- **Linux** /home/MFTZ

2. Extrayez le contenu du package téléchargé:

- **Windows** Sous Windows, utilisez les outils Windows Explorer pour l'extraire.
- **Linux** Sous Linux, procédez à l'extraction et à la décompression comme suit:

```
gunzip V.R.M.F-IBM-MQFA-Redist-LinuxX64.tar.gz
```

puis

```
tar xvf V.R.M.F-IBM-MQFA-Redist-LinuxX64.tar
```

où V.R.M.F est le numéro de version, par exemple 9.3.0.0 ou 9.3.1.0.

Les répertoires suivants sont créés:

- **Windows** **Linux** bin: contient toutes les commandes MFT requises
- **Windows** bin64: contient les bibliothèques requises pour la prise en charge du système d'exploitation Windows 64 bits
- **Windows** **Linux** java: contient les bibliothèques IBM JRE et IBM MQ
- **Windows** **Linux** licenses: contient les fichiers de licence
- **Windows** META-INF: contient des fichiers contenant des informations de signature de code
- **Windows** **Linux** mqft: contient les répertoires ant et lib requis pour la prise en charge de Ant et de la fonction de base MFT
- **Windows** **Linux** swtag: contient le fichier swidtag requis par les gestionnaires de licences pour identifier les installations sur la machine

Que faire ensuite

Vous êtes prêt à configurer un Managed File Transfer Agent. Pour les étapes suivantes, voir [«Création de la configuration initiale pour Redistributable Managed File Transfer Agent»](#), à la page 792.

Depuis IBM MQ 9.3.0, vous pouvez également configurer un Managed File Transfer Logger. Pour les étapes suivantes de configuration du consignateur, voir [«Création de la configuration initiale pour Redistributable Managed File Transfer Logger»](#), à la page 794.

Référence associée

[Erreurs possibles lors de la configuration de Redistributable Managed File Transfer components](#)

Création de la configuration initiale pour Redistributable Managed File Transfer Agent

Vous pouvez configurer un Managed File Transfer Agent pour qu'il se connecte à une configuration IBM MQ existante.

Avant de commencer

Veillez à télécharger et à extraire le contenu du package Redistributable Managed File Transfer Agent . Pour plus d'informations, voir [«Téléchargement et configuration de Redistributable Managed File Transfer components»](#), à la page 790.

Pourquoi et quand exécuter cette tâche

Vous devez d'abord créer l'environnement dont le Redistributable Managed File Transfer Agent a besoin. Vous pouvez ensuite configurer la connectivité avec le gestionnaire de files d'attente qui s'exécute sur le serveur IBM MQ , puis configurer un agent et le gestionnaire de files d'attente d'agent, avant de démarrer et de vérifier l'agent.

Depuis la IBM MQ 9.3.0, l'environnement que vous créez est partagé avec Redistributable Managed File Transfer Logger. Pour plus d'informations, voir [«Création de la configuration initiale pour Redistributable Managed File Transfer Logger»](#), à la page 794.

Procédure

1. Créez l'environnement pour Redistributable Managed File Transfer Agent.

Lorsque vous exécutez la commande **fteCreateEnvironment**, le répertoire de données MFT avec les informations de configuration pour les agents MFT est créé. Vérifiez que vous vous trouvez dans le répertoire bin créé lors de l'extraction du composant Redistributable Managed File Transfer Agent téléchargé. Exécutez ensuite la commande suivante :

- **Windows**

```
fteCreateEnvironment.cmd -d datapath location
```

- **Linux**

```
./fteCreateEnvironment -d datapath location
```

Cette commande utilise les paramètres facultatifs suivants:

-d

Ce paramètre indique l'emplacement du chemin de données dans lequel la configuration MFT est créée, stockée et gérée. Si vous exécutez **fteCreateEnvironment** sans spécifier l'emplacement des données, le répertoire `mftdata` est créé dans l'emplacement où le fichier Redistributable Managed File Transfer Agent est extrait.

Remarque : Si l'agent redistribuable s'exécute en tant que service Windows, la variable d'environnement **BFG_DATA** doit être définie dans l'environnement système pour que le service fonctionne.

-n nom d'installation

Ce paramètre est utilisé pour spécifier le nom d'une installation IBM MQ ou un nom unique.

Voici des exemples de situations dans lesquelles vous pouvez utiliser ce paramètre:

- Si vous souhaitez tester rapidement une nouvelle fonction ou fonction à l'aide du package redistribuable avec la configuration existante dans laquelle les agents ont été configurés pour se connecter au gestionnaire de files d'attente en mode clients uniquement. (Notez que ce paramètre ne s'applique à aucun agent configuré pour se connecter à un gestionnaire de files d'attente en mode liaisons.)

- Si vous effectuez une migration depuis une installation Managed File Transfer standard vers un package Redistributable Managed File Transfer Agent et que vous souhaitez utiliser la même configuration que celle créée par l'installation standard. C'est le cas lorsque le Managed File Transfer standard a été installé mais qu'il se connecte à un gestionnaire de files d'attente d'agent s'exécutant sur une autre machine.

La variable de nom d'installation par défaut est **BFG_INSTALLATION_NAME**.

Pour plus d'informations sur la commande **fteCreateEnvironment**, voir [fteCreateEnvironment](#) (configuration de l'environnement pour Redistributable Managed File Transfer Agent).

Vous pouvez également définir la variable d'environnement *BFG_DATA* avec l'emplacement du chemin de données:

```
BFG_DATA=Datapath location
```

Avant de créer, de démarrer et d'arrêter un agent ou toute autre commande, vous devez vous assurer que la variable *BFG_DATA* est définie sur l'emplacement du chemin de données correct.

2. Configurez la connectivité IBM MQ .

- a) Configurez le gestionnaire de files d'attente de coordination à l'aide de la commande **fteSetupCoordination** .

La commande **fteSetupCoordination** crée l'ensemble requis pour les gestionnaires de files d'attente de coordination et les répertoires requis pour une configuration supplémentaire. Redistributable Managed File Transfer Agent fonctionne en mode client. Vous devez donc fournir des paramètres supplémentaires avec cette commande pour éviter une erreur, car le mode liaisons n'est pas pris en charge.

```
fteSetupCoordination -coordinationQMGr PRMFTDEM02  
-coordinationQMGrHost 9.121.59.233 -coordinationQMGrPort 3002  
-coordinationQMGrChannel SYSTEM.DEF.SVRCONN
```

Pour plus de détails sur l'utilisation de la commande **fteSetupCoordination**, voir [fteSetupCoordination](#). Pour plus d'informations sur la configuration du gestionnaire de files d'attente de coordination, voir «[Configuration du gestionnaire de files d'attente de coordination pour MFT](#)», à la page 833.

- b) Créez et configurez le gestionnaire de files d'attente de commandes:

```
fteSetupCommands -p PRMFTDEM02 -connectionQMGrHost 9.121.59.233  
-connectionQMGrPort 3002 -connectionQMGrChannel SYSTEM.DEF.SVRCONN  
-connectionQMGr PRMFTDEM02 -f
```

Pour plus de détails et les étapes d'utilisation de la commande **fteSetupCommands**, voir [CommandesfteSetup](#): créez le fichier MFT `command.properties`.

3. Créez une définition d'agent MFT pour un noeud final.

```
fteCreateAgent -p PRMFTDEM02 -agentQMGrHost 9.121.59.233  
-agentQMGrPort 3002 -agentQMGrChannel SYSTEM.DEF.SVRCONN  
-agentName AGENT.TRI.BANK -agentQMGr PRMFTDEM02 -f
```

Pour plus d'informations sur l'utilisation de la commande **fteCreateAgent** pour configurer un agent et le gestionnaire de files d'attente d'agent, voir [fteCreateAgent](#).

Remarque : Vous devez utiliser les commandes MQSC affichées dans le résultat de la commande pour définir les objets d'agent sur le gestionnaire de files d'attente de l'agent, faute de quoi les instructions de l'étape «4», à la page 793 ne fonctionneront pas.

Dans les étapes «2», à la page 793 et «3», à la page 793 de chaque agent, vous créez des définitions de file d'attente et de rubrique sur le gestionnaire de files d'attente de l'agent.

4. Démarrez l'agent et vous êtes prêt à transférer des fichiers.

```
fteStartAgent -p PRMFTDEM02 AGENT.TRI.BANK
```

Vous pouvez vérifier le statut de l'agent en exécutant la commande suivante:

`fteListAgents`

Pour plus de détails sur l'utilisation de la commande **fteListAgents**, voir [fteListAgents](#).

Que faire ensuite

Si vous souhaitez configurer le Redistributable Managed File Transfer Logger, suivez les étapes de la rubrique «Création de la configuration initiale pour Redistributable Managed File Transfer Logger», à la page 794.

Concepts associés

«Configuration de Managed File Transfer», à la page 787

Vous pouvez configurer les fonctions d' Managed File Transfer après l'installation.

«Options de configuration de MFT sur Multiplatforms», à la page 787

Managed File Transfer fournit un ensemble de fichiers de propriétés qui contiennent des informations clés sur votre configuration et qui sont nécessaires au fonctionnement. Ces fichiers de propriétés se trouvent dans le répertoire de configuration que vous avez défini lors de l'installation du produit.

Référence associée

fteCreateTransfer: [démarrer un nouveau transfert de fichier](#)

Création de la configuration initiale pour Redistributable Managed File Transfer Logger

Vous pouvez configurer un type de fichier Managed File Transfer Logger pour la connexion à un gestionnaire de file d'attente de coordination en mode client.

Avant de commencer

Veillez à télécharger et à extraire le contenu du package Redistributable Managed File Transfer Agent . Depuis la IBM MQ 9.3.0, ce package inclut également Redistributable Managed File Transfer Logger. Pour plus d'informations, voir «Téléchargement et configuration de Redistributable Managed File Transfer components», à la page 790.

Pourquoi et quand exécuter cette tâche

Redistributable Managed File Transfer Agent et Redistributable Managed File Transfer Logger partagent le même environnement. Une fois que cet environnement a été créé et que la connectivité IBM MQ a été configurée, vous pouvez créer et démarrer le consigneur.

Procédure

1. Assurez-vous que l'environnement partagé pour Redistributable Managed File Transfer Agent et Redistributable Managed File Transfer Logger a été créé comme décrit à l'étape «1», à la page 792 et que la connectivité IBM MQ a été configurée comme décrit à l'étape «2», à la page 793 de «Création de la configuration initiale pour Redistributable Managed File Transfer Agent», à la page 792.

2. Créez un consigneur de fichiers à l'aide de la commande **fteCreateLogger**.

Exemple :

```
fteCreateLogger FILELOGGER -loggerType FILE -loggerQMGr PRMFTDEMO2  
-loggerQMGrHost 9.121.59.233 -loggerQMGrPort 3003 -loggerQMGrChannel SYSTEM.DEF.SVRCONN  
-fileSize 20MB -fileCount 10 -fileLoggerMode CIRCULAR
```

Pour plus d'informations sur l'utilisation de la commande **fteCreateLogger**, voir [fteCreateLogger](#).

3. Démarrez le consigneur à l'aide de la commande **fteStartLogger**.

Pour plus d'informations sur la commande **fteStartLogger**, voir [fteStartLogger](#).

Concepts associés

«Configuration de Managed File Transfer», à la page 787

Vous pouvez configurer les fonctions d' Managed File Transfer après l'installation.

«Options de configuration de MFT sur Multiplatforms», à la page 787

Managed File Transfer fournit un ensemble de fichiers de propriétés qui contiennent des informations clés sur votre configuration et qui sont nécessaires au fonctionnement. Ces fichiers de propriétés se trouvent dans le répertoire de configuration que vous avez défini lors de l'installation du produit.

Windows Linux Mise à niveau Redistributable Managed File Transfer components

Vous pouvez mettre à niveau le Redistributable Managed File Transfer components en téléchargeant un nouveau Redistributable Managed File Transfer package.

Avant de commencer

Pour plus d'informations sur les dispositions du contrat de licence redistribuable pour Redistributable Managed File Transfer Agent et Redistributable Managed File Transfer Logger , voir [IBM MQ Composants redistribuables](#).

Remarque : Advanced Message Security n'est pas pris en charge avec Redistributable Managed File Transfer package.

Pourquoi et quand exécuter cette tâche

Si vous avez déjà installé le Redistributable Managed File Transfer components, vous pouvez le mettre à niveau en téléchargeant un nouveau package redistribuable et en extrayant le contenu dans le même emplacement.

Procédure

1. Téléchargez le package d'agent [IBM MQ redistribuable Managed File Transfer](#) correspondant à votre système d'exploitation à partir de Fix Central.
2. Arrêtez tous les agents Managed File Transfer et le consignateur attend que les commandes Managed File Transfer en cours d'exécution soient terminées.
3. Mettez à jour les fichiers de votre installation existante de Redistributable Managed File Transfer components en extrayant le contenu du nouveau package redistribuable que vous avez téléchargé dans le même répertoire que celui dans lequel vous avez déjà installé Redistributable Managed File Transfer components .

z/OS

Creating an MFT Agent or Logger command data set

You can create a PDSE data set of commands from the Managed File Transfer command template data set for a specific Managed File Transfer Agent or Managed File Transfer Logger for a specific coordination.

About this task

Complete the following steps:

Procedure

1. Make a copy of the MFT command template PDSE library data set SCSQFCMD.
SCSQFCMD must be copied into a new library, for example *prefix.agent*. JCL. You can use an updated version of the SCSQFCMD(BFGCOPY) member with the following replacements:
 - Replace *++supplied-library++* with the fully qualified name of the SCSQFCMD PDSE.

- **z/OS** Replace `++service-library++` with the fully qualified name of the new MFT command PDSE library data set. The `++service-library++` is the output data set for the agent or logger service that is created.
2. For the new MFT command PDSE library data set, edit the member BFGCUSTM, which is a JCL script to customize the commands for the agent or logger. Each variable is specified in the format: `++variable name++`, which you must replace with its required value. For a description of the various JCL variables, see [“z/OS JCL variables” on page 808](#). The BFGSTDIN DD statement defines variables in three categories: Variables, Properties, and Environment. The statement has the following format:

```
[Variables]
variable1=value1
variable2=value2
....
variableN=valueN
[Properties]
property1=property value1
property2=property value2
...
propertyN=property valueN
[Environment]
custom_variable1=value1
custom_variable2=value2
....
custom_variableN=valueN
```

Variables define the set of setup and environment variables that are required for each command.

Properties define overrides for the MFT configuration properties. You can add agent and logger properties as required to customize the agent or logger for your environment. For a list of all properties, see [“Configuration properties files” on page 818](#). This facility is provided to save having to access the MFT configuration properties files, which are maintained as z/OS UNIX System Services files.

Environment defines any additionally required custom environment variables.

3. Submit job BFGCUSTM for the new MFT command PDSE library data set. This job generates the set of JCL commands, as new members of the PDSE, appropriate for the agent or logger. For a full list of the commands, see [“z/OS agent and logger command JCL scripts” on page 811](#).

Job BFGCUSTM updates the library containing the JCL which includes a DD statement with `DISP=OLD`. You must exit the editor after submission to allow the job to execute.

Examine the output job log to check that the JCL script ran successfully. If there are any failures, correct them and submit the BFGCUSTM job again.

The BFGCUSTM JCL script also updates the z/OS UNIX System Services MFT configuration properties files as necessary to keep the files in step. If the configuration defined by the `CoordinationQMgr` property does not exist, warning messages are output and you must run the generated BFGCFGR and BFGCMCR jobs to create the configuration properties files. You must run BFGAGCR for an agent, and BFGLGCRS for a logger edit. If the specified configuration already exists, the configuration is updated with any properties as defined in the BFTCUSTM JCL script.

Related concepts

[“MFT configuration options on z/OS” on page 789](#)

The Managed File Transfer configuration options on z/OS are the same as the options for distributed platforms.

Related tasks

[“Updating an existing MFT Agent or Logger command data set on z/OS” on page 807](#)

You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

z/OS Configuring Managed File Transfer for z/OS

Managed File Transfer for z/OS requires customization to enable the component to operate correctly.

About this task

You need to:

1. Edit a PDSE member to specify configuration data
2. Define the coordination queue manager.
3. Define the command queue manager
4. Configure one or more agents
5. Optionally: configure a logger task to store data in Db2

The sequence of tasks you need to perform is detailed in the following topics.

Related concepts

[“Reviewing the MFT configuration” on page 797](#)

You need to review the configuration of your system before you begin.

Related tasks

[Installing IBM MQ Advanced for z/OS](#)

Reviewing the MFT configuration

You need to review the configuration of your system before you begin.

Managed File Transfer (MFT) requires one or more queue managers to act in the following roles for each defined MFT configuration:

- A coordination queue manager, which maintains information on the status of each agent in the configuration published to a topic on the coordinator.
- One or more command or connection queue managers that act as the entry point to the IBM MQ network for MFT commands.
- One or more agent queue managers that provide the communication between an MFT agent and the IBM MQ network.

Each of the above roles can be performed by a separate queue manager, or you can combine the roles, so that, in the simplest configuration, all roles are performed by a single queue manager.

If you are adding a z/OS queue manager to an existing MFT environment you need to define connectivity between the z/OS queue manager and the other queue managers in the configuration. You can achieve this with manually defined transmission queues, or by the use of clustering.

Each MFT agent communicates with a single queue manager. If multiple agents communicate with the same queue manager, then the agent queue manager will have multiple queues defined for each agent:

- SYSTEM.FTE.COMMAND.*agent_name*
- SYSTEM.FTE.DATA.*agent_name*
- SYSTEM.FTE.REPLY.*agent_name*
- SYSTEM.FTE.STATE.*agent_name*
- SYSTEM.FTE.EVENT.*agent_name*
- SYSTEM.FTE.AUTHAGT1.*agent_name*
- SYSTEM.FTE.AUTHTRN1.*agent_name*
- SYSTEM.FTE.AUTHOPS1.*agent_name*
- SYSTEM.FTE.AUTHSCH1.*agent_name*
- SYSTEM.FTE.AUTHMON1.*agent_name*
- SYSTEM.FTE.AUTHADM1.*agent_name*

Note that you can define generic security profiles, where you use a profile such as SYSTEM.FTE.COMMAND.* , or you can define specific profiles for each agent.

Related concepts

“Avant de commencer à configurer MFT for z/OS” on page 798

La configuration de Managed File Transfer (MFT) utilise les fichiers des ensembles de données z/OS UNIX System Services (z/OS UNIX) et PDSE.

Related reference

[MFT system queues and the system topic](#)

Avant de commencer à configurer MFT for z/OS

La configuration de Managed File Transfer (MFT) utilise les fichiers des ensembles de données z/OS UNIX System Services (z/OS UNIX) et PDSE.

La plupart de la configuration et de l'opération sont effectuées à l'aide de JCL d'un ensemble de données partitionnées étendu (PDSE) et vous devez être familiarisé avec l'utilisation d'un environnement z/OS UNIX .

Vous pouvez accéder à OMVS à partir d' ISPF ou utiliser une session de type Telnet à l'aide de commandes sur votre poste de travail, par exemple Telnet Putty ou SSH.

Si vous utilisez OMVS à partir d' ISPF , vous pouvez utiliser l'éditeur ISPF standard et parcourir les commandes **oedit** et **obrowse**.

Vous devez connaître les commandes z/OS UNIX suivantes

Commande	Fonction
Chemin d'accès à chmod xxx	Modifier les droits d'accès aux fichiers.
Chemin df -k	Indique la quantité d'espace disponible restant dans le système de fichiers. -k indique l'espace disponible en Ko.
chemin du -kt	Indique les tailles des répertoires sous le chemin d'accès. Taille indiquée en Ko.
rechercher le chemin -name xxx	Recherchez le fichier nommé xxxx dans le répertoire du chemin d'accès. xxx est sensible à la casse et peut être similaire à *zzz.
répertoire ls -ltrd	Répertorie des informations sur le répertoire spécifié plutôt que sur les fichiers du répertoire.
ls -ltr chemin	Répertorie les informations sur les fichiers dans le chemin d'accès.
nom de fichier obrowse	Parcourez le nom de fichier.
Nom de fichier oedit	Editez un fichier dans OMVS.

Passez en revue les éléments du tableau suivant et complétez le tableau avec les entrées appropriées pour votre entreprise. Vous avez besoin de ces valeurs lorsque vous éditez le membre [BFGCUSTM](#).

Nom	Exemple de données	Commentaires
ADMIN_JOB1		Carte de travail. Tous les travaux sont générés avec la même carte JCL.

Tableau 42. Paramètres requis pour le membre BFGCUSTM (suite)

Nom	Exemple de données	Commentaires
armELEMENT	Si ARM est utilisé, utilisez la valeur ARM ELEMENT spécifiée dans la règle ARM pour cet agent ou ce consignateur. Si ARM n'est pas utilisé, mettez ce paramètre à blanc ; par exemple, armELEMENT=	
armELEMENTYPE	Si ARM est utilisé, utilisez le paramètre ARM ELEMENTYPE spécifié dans la règle ARM. Par exemple, armELEMENTYPE= SYSBFGAG pour un agent ou armELEMENTYPE= SYSBFGLG pour un consignateur. Si ARM n'est pas utilisé, mettez ce paramètre à blanc ; par exemple, armELEMENTYPE=	
BFG_DATA		Complétez le cas échéant
BFG_GROUP_NAME	MQM	
BFG_JAVA_HOME	/java/java71_bit64_GA/J7.1_64/	
BFG_JVM_PROPERTIES		Complétez le cas échéant
BFG_PROD	/mqm/V9R2M0/mqft	Chemin d'accès complet au répertoire mqft sous le répertoire IBM MQ for z/OS UNIX System Services Components .
BFG_WTO	YES	Pour obtenir un message MFT sur le syslog.
CLEAN_AGENT_PROPS	-trs	Ce paramètre indique les options qui seront utilisées pour nettoyer un agent lors de l'exécution du membre BFGAGCL. Pour plus d'informations sur les valeurs valides pour ce paramètre, voir fteCleanAgent: nettoyage d'un MFT agent .
coordinationQMgr	MQPV	Configuration obligatoire
CHEMIN_DONNÉES_D'ACCÈS		Utilisé dans la migration
Db2_HLQ	SYS2.Db2.Version 10	
DB_PROPS_PATH		Utilisé dans la migration
FTE_CONFIG		Utilisé dans la migration
JOBCARD1		Il s'agit de la carte de travail pour les tâches à exécution longue, les agents et les consignateurs.

Tableau 42. Paramètres requis pour le membre BFGCUSTM (suite)		
Nom	Exemple de données	Commentaires
BIBLIOTHEQUE	SCEN.FTE.JCL	Nom de l'ensemble de données partitionnées étendu MFT. Vous avez besoin d'une copie pour chaque tâche d'agent ou de consignateur.
MQ_HLQ	Qualificatif de haut niveau pour les fichiers IBM MQ . Par exemple, MQM.V920	
MQ_LANG	E	
MQ_PATH	/mqm/V9R2M0	Chemin d'accès complet au répertoire d'installation de IBM MQ for z/OS UNIX System Services Components.
NOM	AGENT1	
CLASSE DE SORTIE	*	
PATH	bin:/usr/bin:/usr/sbin	
productId	ADVANCEDVUE	Ce paramètre permet de définir le type de produit pour lequel l'utilisation de Managed File Transfer doit être enregistrée. Pour plus d'informations sur les valeurs admises pour ce paramètre, voir fteSetProductId: set z/OS SCRT recording product id.
QMGR	MQPV	
SERVICE_TYPE	AGENT ou LOGGER	
TMPDIR	/tmp	Chemin z/OS UNIX accessible en lecture et en écriture pour les fichiers temporaires.

En outre, vous devez passer en revue les variables suivantes et fournir des valeurs si nécessaire:

- coordinationQMgrHôte =
- coordinationQMgrPort =
- Canal coordinationQMgr=
- connectionQMgr=
- connectionQMgrHôte =
- connectionQMgrPort =
- connectionQMgrCanal =

Ces propriétés sont communes à l'AGENT ou au LOGGER.

Remarque : L'hôte, le port et le canal sont requis pour la connexion client mais doivent rester vides pour une connexion de liaison sur la machine locale.

Concepts associés

«Items to check», à la page 801

Ensure that you have enough disk space, a directory for storing data, and that the requisite files exist.

«Editing member BFGCUSTM», à la page 803

You must edit member BFGCUSTM, and enter the values for the parameters that your enterprise uses, before you run the job.

Items to check

Ensure that you have enough disk space, a directory for storing data, and that the requisite files exist.

Check you have enough disk space

Check that you have enough disk space available on the file system where you are going to store the configuration specific files.

If an agent trace is enabled then by default it can use 100 MB of disk space.

The configuration files themselves are small, only a few KB in size.

If you are planning on using two agents and a logger then you need at least 300 MB. You can use the command **df -k path**, where `path` is the location of the installation specific files. This gives the available and total space in KB.

300 MB is 307,200 KB so you should allow for at least 310,000 KB

Create and check the directory for storing Managed File Transfer data

You need a directory for storing the Managed File Transfer (MFT) data.

Check you have enough space in the file system **df -k /var**. This file system should have at least 310,000 KB available.

If you have not created this file system, use the **mkdir** command; for example **mkdir /var/mft**.

Display what permissions users have on this directory, using the command **ls -ltrd /var/mft**.

If the owner or group is not correct, use the command **chown owner:group /var/mft**.

If permissions for the group are not correct, use the following command to give the owner and the group read, write, and execute permissions. Note that the following command also gives all users read and execute permissions **chmod 775 /var/mft**.

Check the files exist and you have access to them

Use the **ls -ltr** command for the files you will be using during customization. For example:

```
ls -ltrd /java/java71_bit64_GA/J7.1_64/bin
```

gives

```
drwxr-xr-x 4 SYSTASK TSouser 8192 Nov 15 2013 /java/java71_bit64_GA/J7.1_64/bin
```

where the `drwxr-xr-x` means

d

This is a directory.

rwX

The owner `SYSTASK` has read, write and execute access to the directory.

r-x

People in the group `TSouser` can read and execute files in the directory.

r-x

Universal access, that is, anyone can read or execute files in the directory.

Check the files specified in:

Table 43. Access required by users to specific files

Path	Access required by users doing the configuration
BFG_JAVA_HOME	Read and execute
/tmp	Read and write
BFG_PROD	Read
BFG_DATA	Write
MQ_PATH	Read

Related concepts

“Avant de commencer à configurer MFT for z/OS” on page 798

La configuration de Managed File Transfer (MFT) utilise les fichiers des ensembles de données z/OS UNIX System Services (z/OS UNIX) et PDSE.

“Common MFT for z/OS configurations” on page 802

An overview of the different Managed File Transfer configurations

Common MFT for z/OS configurations

An overview of the different Managed File Transfer configurations

Managed File Transfer uses agents attached to a queue manager for transferring data.

MFT peut utiliser plusieurs gestionnaires de files d'attente :

- Un ou plusieurs gestionnaires de files d'attente pour transférer les données.
- Un gestionnaire de file d'attente de commandes qui émet des demandes. Par exemple, une demande de démarrage d'un transfert est envoyée à ce gestionnaire de files d'attente et les commandes associées sont routées vers les agents MFT.
- Un gestionnaire de file d'attente de coordination.

Il existe trois configurations Managed File Transfer (MFT) courantes :

1. Un même gestionnaire de files d'attente avec un ou plusieurs agents utilisant des connexions locales. Cette configuration peut être utilisée pour placer le contenu d'un ensemble de données dans des files d'attentes IBM MQ.
2. Un même gestionnaire de files d'attente avec un client MFT sur une machine distribuée utilisant des liaisons client.
3. Deux gestionnaires de files d'attente connectés par des canaux et un ou plusieurs agents sur chaque machine. Ces agents peuvent être des liaisons client ou locales.

Notez les points suivants :

1. MFT est écrit en Java, avec certains scripts shell et JCL pour le configurer et le faire fonctionner.
2. Le statut et l'activité Db2 peuvent être consignés et ces données peuvent être stockées dans des tables Db2.
3. La personne qui configure MFT doit bien connaître z/OS UNIX System Services (z/OS UNIX). Par exemple :
 - La structure de répertoire avec des fichiers dont les noms ressemblent à `/u/userID/myfile.txt2`
 - Les commandes z/OS UNIX, telles que les suivantes :
 - cd** (changement de répertoire)
 - ls** (Liste)
 - chmod** (changement des droits sur les fichiers)
 - chown** (changement de l'appartenance du fichier ou des groupes pouvant accéder au fichier ou au répertoire)

4. Les produits suivants sont requis dans z/OS UNIX pour que MFT puisse être configuré et exécuté :

- Java; par exemple, /java/java71_bit64_GA/J7.1_64/
- IBM MQ V920, par exemple /mqm/V9R2M0.
- Bibliothèques JDBC Db2, si vous souhaitez utiliser Db2 pour l'état et l'historique ; par exemple, /db2/db2v12/jdbc/lib

You need a coordination queue manager. However, you can use the same queue manager to run agents, to process commands, and for coordination. If you are using multiple queue managers, you must pick one to act as the coordinator.

Check your IBM MQ connectivity

If you have an existing MFT coordinator queue manager, you need connectivity between the queue manager where you are doing the configuration, and the coordinating and command queue managers.

Copy SCSQFCMD to create a JCL library

You need to create a JCL library for each agent and logger. The JCL contains the configuration and jobs used to create and run the agent or logger.

For each agent and logger create a copy of the IBM supplied SCSQFCMD library by editing and running the BFGCOPY member.

This library is used to define the configuration for the agent or logger and, after customization, contains jobs that can be used to create the required Managed File Transfer configuration and agent or logger.

You create member BFGCUSTM as part of this process.

Note: If you are familiar with z/OS UNIX commands, you can configure z/OS with the same commands that you use on other platforms.

Related concepts

[“Common MFT for z/OS configurations” on page 802](#)

An overview of the different Managed File Transfer configurations

[“Editing member BFGCUSTM” on page 803](#)

You must edit member BFGCUSTM, and enter the values for the parameters that your enterprise uses, before you run the job.

Editing member BFGCUSTM

You must edit member BFGCUSTM, and enter the values for the parameters that your enterprise uses, before you run the job.

See [Parameters needed for member BFGCUSTM](#), for a list of the parameters requiring specific values.

En outre, vous devez passer en revue les variables suivantes et fournir des valeurs si nécessaire:

- coordinationQMgrHôte =
- coordinationQMgrPort =
- Canal coordinationQMgr=
- connectionQMgr=
- connectionQMgrHôte =
- connectionQMgrPort =
- connectionQMgrCanal =

Ces propriétés sont communes à l'AGENT ou au LOGGER.

Note: L'hôte, le port et le canal sont requis pour la connexion client mais doivent rester vides pour une connexion de liaison sur la machine locale.

If this is the first queue manager in your Managed File Transfer environment, and you want to use the same queue manager for coordination, commands, and running agents, set the values to the local queue manager name.

```
coordinationQMGr=MQPV  
connectionQMGr=MQPV
```

where MQPV is your local queue manager name.

Submit the job, which updates the PDSE, and creates a directory structure under the specified path.

Note that this job requires exclusive use, so you need to stop using the PSDE while the job runs.

Tip: Whenever you submit job BFGCUSTM, the job replaces all the JCL files. You should rename each member you change.

Related concepts

[“Avant de commencer à configurer MFT for z/OS” on page 798](#)

La configuration de Managed File Transfer (MFT) utilise les fichiers des ensembles de données z/OS UNIX System Services (z/OS UNIX) et PDSE.

[“Creating an agent” on page 806](#)

You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

Defining the coordination queue manager

Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

Depending on the configuration that you have chosen, this queue manager is on the local MVS system, or on another machine. In the former case, the connections to it are bindings connections and in the latter case, they are client connections.

After you have run the configuration step successfully there are configured members in the PDSE.

Member BFGCFR defines the coordination queue manager, and this job:

1. Creates a directory structure in the Managed File Transfer (MFT) directory, and creates configuration files.
2. Runs CSQUTIL to define IBM MQ resources.

If the coordination queue manager is on a remote machine then this job step fails.

Member BCFCFCR creates files in z/OS UNIX System Services and creates MQ definitions. This job:

1. Creates an MFT topic,
2. Creates an MFT queue
3. Alters *NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST)* to be *NAMES(SYSTEM.BROKER.DEFAULT.STREAM, SYSTEM.BROKER.ADMIN.STREAM, SYSTEM.FTE)*
4. Performs *ALTER QMGR PSMODE(ENABLED)*

A *DISPLAY NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST)* command is issued before doing the alter. If your NAMLIST is not the default, you should alter your name list to add SYSTEM.FTE to your namelist

Rename member BCFCFCR with your own prefix, for example, CCPCFCR, because re customizing this file replaces it.

Edit this renamed member by inserting the name of your credentials file. For example:

```
%BFGCMD CMD=fteSetupCoordination +  
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>'
```

Save and submit the job. Note that if you need to resubmit the job, you need to add the *-f* option.

When this job runs it lists the IBM MQ resources it creates. You need to protect these resources.

```
DEFINE TOPIC('SYSTEM.FTE') TOPICSTR('SYSTEM.FTE') REPLACE
ALTER TOPIC('SYSTEM.FTE') NPMGDLV(ALLAVAIL) PMSGDLV(ALLAVAIL)
DEFINE QLOCAL(SYSTEM.FTE) LIKE(SYSTEM.BROKER.DEFAULT.STREAM) REPLACE
ALTER QLOCAL(SYSTEM.FTE) DESCR('Stream for MFT Pub/Sub interface')
* Altering namelist: SYSTEM.QPUBSUB.QUEUE.NAMELIST
* Value prior to alteration:
DISPLAY NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST)
ALTER NAMELIST(SYSTEM.QPUBSUB.QUEUE.NAMELIST) +
NAMES(SYSTEM.BROKER.DEFAULT.STREAM+
,SYSTEM.BROKER.ADMIN.STREAM,SYSTEM.FTE)
* Altering PSMODE. Value prior to alteration:
DISPLAY QMGR PSMODE
ALTER QMGR PSMODE(ENABLED)
```

Related tasks

[“Defining the command queue manager” on page 805](#)

You can either use the same queue manager as the coordination and command queue managers, or create a new command queue manager.

Defining the command queue manager

You can either use the same queue manager as the coordination and command queue managers, or create a new command queue manager.

About this task

You must have a command queue manager, however, you can use the same queue manager for the coordination and command queue managers. Otherwise, you need to create a new command queue manager. This can be on the same machine as the coordination queue manager, but does not have to be.

Procedure

1. Rename member BFGCMCR with your own prefix, for example, CCPCMCR.
You must rename BFGCMCR because re-customizing this file replaces it.
2. Edit the renamed member by inserting the name of your credentials file.

For example:

```
%BFGCMD CMD=fteSetupCommands +
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>' +
```

3. Save and submit the job.
Note that if you need to resubmit the job, you need to add the *-f* option.
This queue manager is used for commands such as **ftePingAgent**.
4. Review this member, submit it, and review the output.

What to do next

See [“Creating an agent” on page 806](#) for information on how you create an agent.

Related concepts

[“Defining the coordination queue manager” on page 804](#)

Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

Related tasks

[Configuring MQMFTCredentials.xml](#)

Related reference

[MFT credentials file format](#)

Creating an agent

You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

Review member BFGCUSTM and if you need to use a different credentials file, create one.

Much of the content remains the same from the customization detailed in [“Editing member BFGCUSTM” on page 803](#).

You need to change:

- //SYSEXEC DD DSN=SCEN.FTE.JCL.AGENT1
- LIBRARY to match the agent PDSE
- SERVICE_TYPE=AGENT
- NAME to be the name of the agent (matching the PDSE) JOBCARD
- Change BFG_JVM_PROPERTIES="-Xmx1024M"

Submit this job, remembering that the job requires exclusive access to the data set.

The jobs for the agent all have names of the form *BFGAG**

Rename member *BFGAGCR*. This job updates files in the Managed File Transfer directory and uses CSQUTIL to create agent specific queues in the local queue manager. Specify the name of your credentials file, for example, `-credentialsFile //' SCEN.FTE.JCL.VB(CREDOLD)`. If you do not specify the name, the job to start the agent does not use a credentials file.

Check the output to ensure that the process has run successfully.

Tip: Copy the path name of the *agent.properties* file from the output of the job to a member in the PDSE for the agent.

For example, copy `/u/userid/fte/wmqmft/mqft/config/MQPA/agents/AGENT1/agent.properties` into member AGENT.

This is useful if you need to display the properties file, and add the line `/u/userid/fte/wmqmft/mqft/logs/MQPA/agents/AGENT1/logs`.

This is where trace files are stored.

Related concepts

[“Defining the coordination queue manager” on page 804](#)

Managed File Transfer requires a queue manager to be created that acts as the coordination queue manager.

[“Using the agent” on page 806](#)

How you use various commands to ensure that the agent is working correctly.

Related tasks

[“Defining the command queue manager” on page 805](#)

You can either use the same queue manager as the coordination and command queue managers, or create a new command queue manager.

Using the agent

How you use various commands to ensure that the agent is working correctly.

Start the agent

Rename member BFGAGST, review the member, and submit the job.

If this works you receive message BFGAG0059I: The agent has been successfully started.

Display the active agent(s)

Rename member BFGAGLI, review the member and submit the job which uses the coordinating queue manager.

You must resolve any connectivity problems

Ping the agent to check it is working

Rename member BFGAGPI, review the member and submit the job which uses the command queue manager.

You must resolve any connectivity problems

Carry out a test transfer

See [“Performing a verification transfer” on page 813](#) for further information.

Stop the agent

Rename member BFGAGSP, review the member and submit the job.

Restart the agent using the member BFGAGST.

Related concepts

[“Creating an agent” on page 806](#)

You need to copy the PDSE to make the agent-specific PDSE, for example *user.MFT.AGENT1*. Copy the PDSE from a previous agent or logger configuration, if they exist. If this is your first configuration, copy the PDSE supplied with MFT.

Updating an existing MFT Agent or Logger command data set on z/OS

You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

Procedure

1. Edit the BFGCUSTM JCL script member and update variables and properties in the BFGSTDIN DD statement.

If you want to remove a property that was previously defined, set its value to blank, instead of removing the entry. When the BFGCUSTM JCL script is run, the specified properties are applied as an update to the actual agent and logger z/OS UNIX System Services properties files; setting a property to a blank value indicates that the property is to be removed

2. Submit job BFGCUSTM. This job generates the set of JCL commands again, appropriate for the agent or logger. For a full list of the commands, see [“z/OS agent and logger command JCL scripts” on page 811](#). Examine the output job log to check that the JCL script ran successfully. If there are any failures, correct them and submit the BFGCUSTM job again.

Results

You can modify the generated JCL scripts and add your own logic. However, be careful when you run BFGCUSTM again because you might overwrite the custom logic.

Related concepts

[“MFT configuration options on z/OS” on page 789](#)

The Managed File Transfer configuration options on z/OS are the same as the options for distributed platforms.

Related tasks

[“Creating an MFT Agent or Logger command data set” on page 795](#)

You can create a PDSE data set of commands from the Managed File Transfer command template data set for a specific Managed File Transfer Agent or Managed File Transfer Logger for a specific coordination.

z/OS JCL variables

You can use substitution values, JCL variables, and configuration properties in the BFGCUSTM script.

The following table lists the substitution values for the BFGCUSTM JCL script in an MFT command PDSE library data set. You must replace these substitution values with suitable values before you submit the BFGCUSTM job.

Substitution variable	Value
++library++	The data set name of the containing MFT command PDSE library.
++bfg_java_home++	The location of your Java installation.
++mq_path++	The path to the IBM MQ for z/OS UNIX System Services Components directory. For example, /mqm/V9R2M0. This is used to give the full path to the MFT installation, for example, /mqm/V9R2M0/mqft.

The following table describes the environment variables for the BFGSTDIN DD statement for the BFGCUSTM JCL script, in an MFT command PDSE library data set (in the [Variables] section). You must replace all variables that are specified with substitution values (that is, values enclosed in two plus signs, ++) with suitable values before you submit the BFGCUSTM job.

Environment variable	Value
LIBRARY	The data set name of the containing MFT command PDSE library.
TMPDIR	z/OS UNIX System Services directory for temporary files.
BFG_PROD	The full path to the mqft directory under the IBM MQ for z/OS UNIX System Services Components directory; For example: /mqm/V9R2M0/mqft.
BFG_DATA	The location of the data directory for Managed File Transfer for z/OS, which is the path to <i>DATA_DIR</i> .
BFG_JAVA_HOME	The location of your Java installation.
BFG_JVM_PROPERTIES	Optional. Sets a value for the BFG_JVM_PROPERTIES environment variable. These properties are passed to the Java virtual machine.

Table 45. Environment variables (continued)

Environment variable	Value
BFG_GROUP_NAME	<p>The mqm file group is typically associated with MFT configuration data files and commands. Consequently, all users who are members in the mqm group can access and make changes to the MFT configuration. For more information, see File system permissions for MFT in IBM MQ.</p> <p>For a z/OS system, a file group is a z/OS UNIX System Services (z/OS UNIX) filesystem entity, and the mqm file group is not necessarily defined. You can associate a z/OS UNIX filesystem group for MFT configuration data files by using the BFG_GROUP_NAME environment variable. For example, at the z/OS UNIX shell prompt use:</p> <pre data-bbox="862 680 1472 758">export BFG_GROUP_NAME=FTEGB</pre> <p>which defines a group <i>FTEGB</i> to be associated with any subsequently created configuration files for the current z/OS UNIX session.</p> <p>You can set BFG_GROUP_NAME to a blank value, or remove it.</p> <p>Note: When running BFGCUSTM for the first time, if the MFT configuration is to be used by multiple user IDs, it is important that BFG_GROUP_NAME is set to a group accessible to all required user ID's. If BFGCUSTM is run again, then BFG_GROUP_NAME must not be changed (otherwise, the z/OS UNIX group file permissions for all files and directories in the directory referenced by BFG_DATA must also be changed to reflect the new BFG_GROUP_NAME setting).</p>
BFG_WTO	<p>z/OS logging is enabled when BFG_WTO is set to YES, ON, or TRUE. This controls whether messages that are written to the agent event log are also written to the z/OS operator log facility, which allows easier access for automation products when you run an agent from JCL. The routing code is Programmer Information (11) and the descriptor code is Informational (12).</p>
SERVICE_TYPE	<p>Specifies whether the MFT command library is for an agent or logger. The valid values are AGENT or LOGGER.</p>
NAME	<p>The name of the agent or logger for the SERVICE_TYPE value.</p>
QMGR	<p>The name of the local queue manager that is associated with the agent or logger for the SERVICE_TYPE value.</p>

<i>Table 45. Environment variables (continued)</i>	
Environment variable	Value
OUTPUT_CLASS	The output class for SYSOUT data sets. Defaults to * which requests the same output class as the MSGCLASS parameter from the job statement.
MQ_PATH	The path to the IBM MQ for z/OS UNIX Components directory.
MQ_HLQ	The high-level qualifier for IBM MQ data sets.
MQ_LANG	The language that is required.
DB2_HLQ	Optional. High-level qualifier for Db2 data sets.
JOBCARD1	Header line 1 for a JCL command job.
JOBCARD2	Header line 2 for a JCL command job.
JOBCARD3	Header line 3 for a JCL command job.
ADMIN_JOB1	Header line 1 for an admin job.
ADMIN_JOB2	Header line 2 for an admin job.
ADMIN_JOB3	Header line 3 for an admin job.
FTE_CONFIG	Existing MFT configuration for migration. Set to a blank value if migration is not required.
CREDENTIAL_PATH	Path to credentials file for migration, for example /u/user1/agent3. Required for migration commands BFGAGMG and BFGLGMG JCL scripts only. Set to a blank value if migration is not required. Note also that
DB_PROPS_PATH	Specifies the database logger properties file for migration. This option is required only if the properties file does not use the following default name and path: config_directory/coordination_qmgr/ databaselogger.properties. Set to a blank value if migration is not required.

The following table describes the mandatory MFT configuration properties for the BFGSTDIN DD statement for the BFGCUSTM JCL script in an MFT command PDSE library data set. You must replace properties specified with substitution values (that is, values enclosed in two plus signs, ++) with a suitable non-blank value before you submit the BFGCUSTM job. These properties define overrides for the MFT configuration properties. You can add agent and logger properties to customize agents or loggers for your environment. For a list of all properties, see “Configuration properties files” on page 818.

<i>Table 46. Mandatory configuration properties for the BFGSTDIN DD statement</i>	
Property	Value
coordinationQMGr	The name of the coordination queue manager for the configuration that the agent or logger is associated with.
coordinationQMGrHost	Optional. Host name of the system that the coordination queue manager is running on. If you leave the value for this property blank, a bindings mode connection is assumed.

Table 46. Mandatory configuration properties for the BFGSTDIN DD statement (continued)

Property	Value
coordinationQMGrPort	Optional. Port number that the coordination queue manager is listening on. This parameter is used only if you also specify a non-blank value for the coordinationQMGrHost property.
coordinationQMGrChannel	Optional. Channel to use to connect to the coordination queue manager. This parameter is used only if you also specify a non-blank value for the coordinationQMGrHost property.
connectionQMGr	The name of the command queue manager for the configuration that the agent or logger is associated with.
connectionQMGrHost	Optional. Host name of the system that the command queue manager is running on. If you leave the value for this property blank, a bindings mode connection is assumed.
connectionQMGrPort	Optional. Port number that the command queue manager is listening on. This parameter is used only if you also specify a non-blank value for the connectionQMGrHost property.
connectionQMGrChannel	Optional. Channel to use to connect to the command queue manager. This parameter is used only if you also specify a non-blank value for the connectionQMGrHost property.

z/OS agent and logger command JCL scripts

The set of JCL commands available in an MFT command PDSE library data set.

Table 47. JCL commands available in an MFT command PDSE library data set

Member	Description or fte command line command
BFGCOPY	Job to create a copy of this library
BFGCUSTM	Job to customize this library for agent or logger
BFGZCFRC	fteSetupCoordination
BFGZCMCR	fteSetupCommands: create the MFT command.properties file
BFGZAGCR	fteCreateAgent . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGLGCRS	fteCreateLogger . Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZAGST	fteStartAgent . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGAGSTP	fteStartAgent procedure. Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZAGPI	ftePingAgent . Created only when you set the SERVICE_TYPE variable to AGENT.

Table 47. JCL commands available in an MFT command PDSE library data set (continued)

Member	Description or fte command line command
BFGZAGSP	<code>fteStopAgent</code> . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGST	<code>fteStartLogger</code> . Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGLGSTP	fteStartLogger procedure. Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZLGSP	<code>fteStopLogger</code> . Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZAGSH	<code>fteShowAgentDetails</code> . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGSH	<code>fteShowLoggerDetails</code> . Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZCFDF	<code>fteChangeDefaultConfigurationOptions</code>
BFGZAGCL	<code>fteCleanAgent</code> . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZAGDE	<code>fteDeleteAgent</code> . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGDE	<code>fteDeleteLogger</code> . Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGZPRSH	<code>fteDisplayVersion</code>
BFGZAGLI	<code>fteListAgents</code> . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZMCLI	<code>fteListMonitors</code>
BFGZSTLI	<code>fteListScheduledTransfers</code>
BFGZTMLI	<code>fteListTemplates</code>
BFGXCROB	fteObfuscate sample
BFGZRAS	fteRAS
BFGZAGTC	<code>fteSetAgentTraceLevel</code> . Created only when you set the SERVICE_TYPE variable to AGENT.
BFGZLGTC	<code>fteSetLoggerTraceLevel</code> . Created only when you set the SERVICE_TYPE variable to LOGGER.
BFGXPRAN	fteAnt sample
BFGXTRCA	fteCancelTransfer sample
BFGXMNCR	fteCreateMonitor sample
BFGXTMCR	fteCreateTemplate sample
BFGXTRCR	fteCreateTransfer sample
BFGXMNDE	fteDeleteMonitor sample
BFGXSTDE	fteDeleteScheduledTransfer sample
BFGXTMDE	fteDeleteTemplate sample

Notes:

- The JCL, for commands that create MQSC or reference delete scripts, asks you to run a script, but the script has already been run by the job.
- BFGZRAS creates the BFGRAS member when the BGCUSTM job is run.

Performing a verification transfer

How you carry out a transfer to check that the product is working correctly.

Rename and edit member BFGTRCRS.

1. Add a /* before the %BFGCMD CMD=fteCreateTransfer -h
2. Remove the other comments in the member.
3. Specify the current agent name for -sa and -da
4. Save the JCL
5. Submit the JCL

This JCL connects to the command queue manager.

Configuration d'une tâche de journalisation

La tâche de journalisation doit être exécutée sur la même image que le gestionnaire de file d'attente de coordination. Vous pouvez vous connecter à Db2.

Création d'une tâche de journalisation

Copiez l'ensemble de données partitionnées étendu pour que l'ensemble de données partitionnées étendu soit spécifique au consignateur. Par exemple, user.MFT.LOGGER.

Si vous devez utiliser un autre fichier de données d'identification, créez-en un. Voir [Configuration de MQMFTCredentials.xml](#) sur z/OS.

Réviser le membre `BFGCUSTM`. Notez qu'une grande partie du contenu reste identique à celui de la personnalisation précédente.

Cependant, vous devez:

- Modifiez // SYSEXEC DD DSN=SCEN.FTE.JCL....
- Modifiez LIBRARY pour qu'il corresponde à l'ensemble de données partitionnées étendu de l'agent
- Remplacez QMGR par le nom du gestionnaire de files d'attente de coordination
- Make SERVICE_TYPE=XX_ENCODE_CASE_ONE consignateur
- Remplacez NAME par le nom du consignateur (correspondant au PDSE)
- Passez en revue la carte de travail et modifiez le nom de travail de sorte qu'il soit différent des noms de travail des agents.
- Vérifiez BFG_JVM_PROPERTIES = "-Xmx1024M"

Si vous utilisez le consignateur Db2, il est utile de créer un fichier afin de pouvoir capturer des traces Db2 pour identifier les problèmes liés à Db2.

Le nom du fichier est spécifié dans les propriétés JVM, où le fichier de propriétés de trace JDBC a un contenu tel que:

```
db2.jcc.traceDirectory=/u/johndoe/fte
db2.jcc.traceFile=jccTrace1
db2.jcc.traceFileAppend=false
# turn on all traces
# db2.jcc.traceLevel=-1
# turn off all traces
db2.jcc.traceLevel=0
```

Définition de deux propriétés JVM

```
BFG_JVM_PROPERTIES=-Ddb2.jcc.propertiesFile=/u/.../sql.properties  
-Ddb2.jcc.ssid=DBCA
```

Où `/u/.../sql.properties` est le nom de votre fichier de propriétés de trace Db2 et `DBCA` est le nom de votre sous-système Db2 .

Soumettez ce travail, en notant que le travail requiert un accès exclusif au fichier. Les travaux de l'agent portent tous des noms tels que `BFGLG*`.

Consignation dans des fichiers

Pour plus d'informations sur la connexion à Db2, voir [«Création d'une tâche de consignation lors de la consignation dans Db2»](#), à la page 815

Renommez le membre `BFGLGCRS`. Ce travail met à jour les fichiers du répertoire Managed File Transfer (MFT) et utilise `CSQUTIL` pour créer des files d'attente spécifiques à l'agent dans le gestionnaire de files d'attente local.

Le fichier d'origine contient la commande `%BFGCMD CMD=fteCreateLogger -h` qui répertorie la syntaxe de la commande.

Pour créer la tâche de consignateur, mettez en commentaire le `%BFGCMD CMD=fteCreateLogger -h` en plaçant `/*` devant l'instruction, en vous assurant que la colonne 1 est vide.

Supprimez les commentaires de la deuxième commande et configurez les instructions. Exemple :

```
%BFGCMD CMD=fteCreateLogger +  
-p MQPH +  
-loggerMgr MQPH +  
-loggerType FILE +  
-fileLoggerMode circular +  
-fileSize 5MB +  
-fileCount 5 +  
-p MQPH +  
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>'  
LOGGER
```

Vérifiez la sortie pour vous assurer qu'elle a été correctement traitée.

Conseil : Copiez le nom de chemin du fichier `logger.properties` de la sortie du travail vers un membre du PDSE de l'agent.

Par exemple, copiez dans le membre `APATH`

```
/u/user_ID/fte/wmqmft/mqft/config/MQPH/loggers/LOGGER/logger.properties
```

Ceci est utile si vous devez afficher le fichier de propriétés.

Ajoutez le répertoire à ce fichier:

```
/u/user_ID/fte/wmqmft/mqft/logs/MQPH/loggers/LOGGER/
```

Si vous vous connectez à un fichier, les fichiers journaux sont stockés dans ce répertoire, par exemple `LOGGER0-20140522123654897.log`.

Les fichiers de trace se trouvent dans le sous-répertoire `log`, par exemple

```
/u/user_ID/fte/wmqmft/mqft/logs/MQPH/loggers/LOGGER/logs
```

Vous pouvez maintenant [démarrer la tâche de journalisation](#).

Création d'une tâche de consignation lors de la consignation dans Db2

Renommez le membre BFGLGCRS.

Ce travail met à jour les fichiers dans le répertoire MFT et utilise CSQUTIL pour créer des files d'attente spécifiques à l'agent dans le gestionnaire de files d'attente local.

Vous devez savoir:

Nom Db2	Exemple
-dbName databaseName	Vous pouvez l'obtenir à partir de la valeur d'emplacement dans le message DSNL004I pour votre sous-système Db2
-dbDriver filePath	Par exemple /db2/db2v10/jdbc/classes/db2jcc.jar
-dbLib filePath	Par exemple /db2/db2v10/jdbc/lib/libdb2jcc2zos_64.so

Éditez le fichier. Le fichier d'origine contient la commande %BFGCMD CMD=fteCreateLogger -h qui répertorie la syntaxe de la commande.

Supprimez les commentaires de la deuxième commande et configurez les instructions. Par exemple

```
%BFGCMD CMD=fteCreateLogger +
-p MQPH +
-loggerQMgr MQPH +
-loggerType DATABASE +
-dbType DB2 +
-databaseName DSNDBCP +
-dbDriver /db2/db2v10/jdbc/classes/db2jcc.jar +
-dbLib /db2/db2v10/jdbc/lib/ +
-credentialsFile //'<MFTCredentialsDataSet(MemberName)>' +
LOGGER
```

Pour créer la tâche de consignateur, mettez en commentaire le %BFGCMD CMD=fteCreateLogger -h en plaçant /* devant l'instruction, en vous assurant que la colonne 1 est vide.

Soumettez le travail et vérifiez la sortie pour vous assurer qu'il a été correctement traité.

Conseil : Copiez le nom de chemin du fichier logger.properties de la sortie du travail vers un membre du fichier PDSE des agents.

Par exemple, copiez dans le membre APATH:

```
/u/user_ID/fte/wmqmft/mqft/config/MQPH/loggers/LOGGER/logger.properties into member USS
```

Ceci est utile si vous devez afficher le fichier de propriétés

Les fichiers de trace se trouvent dans le sous-répertoire log, par exemple:

```
/u/user_ID/fte/wmqmft/mqft/logs/MQPH/loggers/LOGGER/logs
```

Création de tables Db2

Vous devez créer les tables Db2. Les définitions se trouvent dans le fichier z/OS UNIX System Services mqft/sql/ftelog_tables_zos.sql.

Créez un membre Db2 dans votre PDSE. Éditez ce membre et utilisez la commande COPY sur la ligne de commande. Effectuez une copie à partir du fichier de définitions z/OS UNIX System Services.

Étant donné que les exigences spécifiques au site peuvent varier considérablement, ce fichier ne spécifie que les structures de base des tables et un espace table où elles seront situées.

L'espace table est spécifié par le script SQL pour s'assurer qu'il est créé à l'aide d'un pool de mémoire tampon avec une taille de page suffisante pour contenir les lignes de tables les plus volumineuses possibles. Notez que les attributs tels que les emplacements LOB, etc., ne sont pas spécifiés.

Votre administrateur de base de données peut vouloir modifier une copie de ce fichier afin de définir ces attributs liés aux performances.

Ce fichier suppose également le nom de schéma par défaut FTELOG, le nom d'espace table par défaut FTELOGTSet le nom de base de données FTELOGDB. Vous pouvez modifier ces noms si nécessaire, pour qu'ils correspondent à une base de données existante et à toute convention de dénomination locale, en suivant le processus décrit dans les commentaires au début du fichier.

Important : Utilisez des fonctions en ligne telles que **SPUFI** pour exécuter les commandes, car le fichier contient des commentaires, et les programmes de traitement par lots tels que **DSNTINAD** n'acceptent pas les commentaires.

Pour plus d'informations, voir [Exécution de SQL à l'aide de SPUFI](#) . En outre, CSQ45STB dans SCSQPROC comporte un exemple de JCL que vous pouvez personnaliser pour exécuter les commandes Db2 SELECT.

Démarrage de la tâche du consignateur

Renommez, passez en revue et soumettez le membre BFGLGST. Vous devriez recevoir le message BFGDB0023I: Le consignateur a terminé les activités de démarrage et est maintenant en cours d'exécution.

Opérations du consignateur

Pour afficher le statut du consignateur, renommez, révisez et soumettez le membre BFGLGSH

Pour arrêter le consignateur, renommez, révisez et soumettez le membre BFGLSPG.

Environment variables for MFT on z/OS

If you are running commands direct from the z/OS UNIX System Services (z/OS UNIX) environment, or your own JCL scripts, after customization and configuration you must set a number of environment variables before running the configuration and administration scripts provided by Managed File Transfer. You must set these variables for each user and in each environment that the scripts will be invoked from.

To avoid conflicts with other products, you can choose to create a `.wmqfterc` script in your home directory. The `.wmqfterc` script is then invoked by each of the Managed File Transfer scripts and you can use this script to provide custom environment settings for Managed File Transfer.

There is also one optional environment variable, `BFG_WTO`, that you can set to send messages to the operator log when running agents from JCL.

<i>Table 49. Required z/OS environment variables</i>	
Environment variable	Value
BFG_JAVA_HOME	The location of your Java installation. For more information about the levels of Java supported, see System Requirements for IBM MQ .
BFG_DATA	The location of the data directory for Managed File Transfer for z/OS. This is the path to <code>DATA_DIR</code> .

Table 49. Required z/OS environment variables (continued)

Environment variable	Value
STEPLIB	<p>Must include the following IBM MQ data sets:</p> <ul style="list-style-type: none"> • SCSQAUTH • SCSQANLE • SCSQLOAD <p>If you want to run the database logger component on a z/OS system, STEPLIB must also include the following Db2 data sets in the order shown:</p> <ul style="list-style-type: none"> • SDSNEXIT • SDSNLOD2 • SDSNLOAD

The following is an example .profile that correctly configures the environment variables for Managed File Transfer:

```
STEPLIB=MQM.V920.SCSQAUTH:MQM.V920.SCSQANLE:MQM.V920.SCSQLOAD
PATH=/u/ftuser/bin:/u/ftuser/J7.0/bin:/bin:/usr/bin:/u/ftuser/extras/bin:/bin:$PATH
BFG_JAVA_HOME=/u/ftuser/J7.0
BFG_DATA=/u/ftuser/DATA_DIR
export PATH STEPLIB BFG_JAVA_HOME BFG_DATA
```



Attention: The LIBPATH environment variable is no longer needed when calling **fte*** commands from a z/OS UNIX environment, and should be removed from any existing .wmqfterc script

Optionally, you can also set the following environment variables:

Table 50. Optional z/OS environment variable

Environment variable	Value
BFG_WTO	<p>One of the following values will enable BFG_WTO :</p> <ul style="list-style-type: none"> • YES • ON • TRUE <p>One of the following values will disable BFG_WTO. These values are not case sensitive.</p> <ul style="list-style-type: none"> • NULL • NO • OFF • FALSE <p>Enables z/OS logging. By default, this environment variable is disabled.</p> <p>Messages that are written to the agent event log are also written to the z/OS operator log facility, which allows easier access for automation products when you run an agent from JCL. The routing code is Programmer Information (11) and the descriptor code is Informational (12).</p>

Table 50. Optional z/OS environment variable (continued)

Environment variable	Value
BFG_GROUP_NAME	<p>The mqm file group is typically associated with Managed File Transfer configuration data files and commands. Consequently, all users who are members of the mqm group can access, and make changes to the Managed File Transfer configuration. For more information, see File system permissions for MFT in IBM MQ.</p> <p>For a z/OS system, a file group is a z/OS UNIX filesystem entity, and the mqm file group is not necessarily defined. You can define an alternative, existing z/OS UNIX filesystem group for Managed File Transfer configuration data files by using the BFG_GROUP_NAME environment variable. For example, at the z/OS UNIX shell prompt:</p> <pre data-bbox="862 720 1463 793">export BFG_GROUP_NAME=FTEGB</pre> <p>which defines group FTEGB to be associated with any subsequently created configuration files for the current z/OS UNIX session.</p> <p>You can set BFG_GROUP_NAME to a blank value, or remove it.</p>

Configuration properties files

A summary of the properties that are used in Managed File Transfer.

- [The MFT coordination.properties file](#)
- [The MFT command.properties file](#)
- [The MFT agent.properties file](#)
- [Logger configuration properties file](#)

Configuring MFT for the z/OS Automatic Restart Manager (ARM)

Managed File Transfer is an ARM enabled application.

Before you begin

For more information about enabling ARM, and defining ARM policies for your system, see [Using the z/OS Automatic Restart Manager \(ARM\)](#).

If you want to use the MFT DB Logger ability to automatically restart and reconnect to a Db2 database, ARM is the only supported restart manager available.

About this task

Using ARM, agents and loggers can be configured for restart by setting the agent/logger properties armELEMENTYPE, and armELEMENT. Property armELEMENTYPE defines the type of ARM element and property armELEMENT is the name of the element that ARM is to register:

- You can set the agent ELEMTYPE to SYSBFGAG, and armELEMENT can be set to correspond with the agent name.
- You can set the logger ELEMTYPE to SYSBFGLG, and armELEMENT can be set to correspond with the logger name.

Note: Agents and loggers that are configured for restart by ARM can only be successfully run from a batch job or a started task. Attempts to start the agent or logger from the z/OS UNIX System Services command line directly will fail with an ARM error reason code.

Example

The following example of a restart policy defines agent BFGFT7CAG1 as being dependant on queue manager FT7C:

```
RESTART_ORDER
  LEVEL(3)
  ELEMENT_TYPE(SYSBFGAG,SYSBFGLG)

RESTART_GROUP(GROUP7C)
  ELEMENT(SYSMQMGRFT7C)
  ELEMENT(BFGFT7CAG1)
  RESTART_ATTEMPTS(3,300)
```

Example: Creating JCL for Managed File Transfer agents on z/OS

Use this information to generate some JCL that can be used to create and start an agent on IBM MQ for z/OS.

Copy the sample library

Carry out the following procedure:

1. Make a copy of the library SCSQFCMD (see [“Copy SCSQFCMD to create a JCL library”](#) on page 803) by opening the library.

The majority of the members, those that start with BFGX, BFGY, or BFGZ, are templates that you use to generate the customized JCL for the agent later on.

The important member is BFGCOPY.

2. Open BFGCOPY and replace:

++supplied_library++

with the name of the SCSQFCMD library that was installed as part of the product.

++service-library++

with the name of the library that you want to use for your agent (the target library).

3. Submit the job and you have a new library that you can use.

Edit BFGCUSTM

Carry out the following procedure:

1. Open the new library so that you can edit the BFGCUSTM member (see [“Editing member BFGCUSTM”](#) on page 803)
2. Modify all of the parameters in the member that are enclosed within ++ characters, and replace them with the appropriate values. For example, change:

++mq_path++

The path to the z/OS UNIX System Services (z/OS UNIX) Components directory. For example, /mqm/V9R2M0.

Note: There are three instances of this variable to replace.

++bfg_data++

To be the z/OS UNIX directory where your IBM MQ Managed File Transfer for z/OS configuration is to be stored.

++service_type++

To the word AGENT

++agent_name++

To be the name of your agent

Notes:

1. Some of the entries, such as ++options++ required for the CLEAN_AGENT_PROPS, are not needed and so you should remove these.
2. See [“Avant de commencer à configurer MFT for z/OS”](#) on page 798 for a complete list of all of the parameters in the BFGCUSTM member, along with a description of what values they should have.

Submit the BFGCUSTM JCL

Carry out the following procedure:

1. Submit the job.
2. Exit the library in ISPF.

This is necessary because the BFGCUSTM job is updating the library, and cannot do that while the library is open.

3. When the job completes look at the joblog.

You will see a number of messages, indicating that new members have been created within the library.

Each of these members contains JCL that can be used to perform specific tasks for your agent. See [“z/OS agent and logger command JCL scripts”](#) on page 811 for a list of these members, along with the IBM MQ Managed File Transfer commands that they correspond to.

Submit BFGAGCR to create the agent

The new member BFGAGCR contains some JCL that [creates an agent](#) by invoking the **fteCreateAgent** command.

Carry out the following procedure:

1. Open up member BFGAGCR.

You should see that BFGAGCR has been populated with the name of your:

- Agent
- Agent queue manager
- The coordination queue manager for the MFT topology

2. Submit member BFGAGCR.

When the member runs, it:

- Creates the required configuration files for your agent.
- Connects to the agent queue manager, and creates the system queues that the agent needs, using CSQUTIL.
- Registers the agent with the coordination queue manager.

Start the agent by submitting BFGAGST

Carry out the following procedure:

1. Submit the BFGAGST member. See [using the agent](#) for various commands that show you that the agent is working correctly.

2. When the job completes, check the joblog contains the following messages:

```
BFGAG0058I: The agent has successfully initialized.  
BFGAG0059I: The agent has been successfully started.
```

which means that your agent is up, running, and ready to perform managed transfers.

Moving an MFT agent to a new z/OS LPAR

It is sometimes necessary to move an IBM MQ Managed File Transfer for z/OS agent from one LPAR to another, while keeping the agent in the same IBM MQ Managed File Transfer topology with the same coordination and command queue managers. The steps needed to do this depend on how the agent being migrated was originally created.

About this task

Move your IBM MQ Managed File Transfer for z/OS agent in one of the following ways:

- If the agent was originally created using a customized version of the SCSQFCMD library, use the library to recreate it on a new LPAR.
- If the agent was originally created by running z/OS UNIX System Services (z/OS UNIX) commands, use the commands to recreate it on a new LPAR.

Note:

Scheduled transfers and transfer templates are stored on the coordination queue manager for an IBM MQ Managed File Transfer topology. This task assumes that the coordination queue manager is not part of the movement work. In this case, any scheduled transfers and transfer templates associated with the agent being moved remain on the existing coordination queue manager after the move is completed.

Procedure

- Move an agent created using a customized version of the SCSQFCMD library.

If the agent was created using a customized version of the SCSQFCMD library, you can use that library to recreate the IBM MQ Managed File Transfer for z/OS environment, and the agent configuration on the new LPAR. To do this, complete the following steps:

1. Copy the customized version of the library from the original LPAR to the new LPAR.
2. Edit the BFGCUSTM member in the customized version of the library on the new LPAR, and make sure that the parameter values are still valid.
3. Run the BFGCUSTM member on the new LPAR, to create all of the JCL needed to configure the environment and create the agent.
4. Run the BFGCFR member to define the coordination queue manager to be used by the agent on the new LPAR, and create the directory structure needed to store the IBM MQ Managed File Transfer configuration.
5. Next, run the BFGCMCR member, to define the command queue manager to be used by the agent on the new LPAR.
6. Run the BFGAGCR member to recreate the agent and its configuration.
7. Ensure that the system queues used by the agent exist on the queue manager for that agent.

If the agent being moved has resource monitors associated with it, you need to recreate the monitors on the new agent. To do this, complete the following steps:

1. On the original LPAR, run the BFGMCLI member to export the definitions for the resource monitor associated with the original agent to XML files.
2. Copy the XML files containing the resource monitor definitions to the new LPAR.

3. Use the BFGMNCRS member in the SCSQFCMD library on the new LPAR to import the resource monitor definitions stored in the XML files. This results in the monitors being created on the new agent.
- Move an agent created by running commands in z/OS UNIX.

If the agent was originally created by running z/OS UNIX commands, you can use commands to recreate the agent on a new LPAR. To do this, complete the following steps:

1. Run the `fteSetupCoordination` command on the new LPAR, to define the coordination queue manager to be used by the agent, and create the directory structure needed to store the IBM MQ Managed File Transfer configuration.
2. Run the `fteSetupCommands` command to define the command queue manager to be used by the agent on the new LPAR.
3. Run the `fteCreateAgent` command to recreate the agent and its configuration.
4. Ensure that the system queues used by the agent exist on the queue manager for that agent.

If the agent being moved has resource monitors associated with it, you need to recreate the monitors on the new agent. To do this, complete the following steps:

1. On the original LPAR, run the `fteListMonitors` command, specifying the `-ox` parameter, to export the definitions for the resource monitor, associated with the original agent, to XML files.
2. Copy the XML files containing the resource monitor definitions to the new LPAR.
3. Run the `fteCreateMonitor` command on the new LPAR, specifying the `-ix` parameter, to import the resource monitor definitions stored in the XML files. This results in the monitors being created on the new agent.

Planning your MFT infrastructure with IBM MQ for z/OS queue sharing groups

You need to consider the following, if you are using IBM MQ Managed File Transfer (MFT), when one or more of the agents, command or coordination queue managers are part of an IBM MQ for z/OS queue sharing group.

See [MFT topology overview](#) for a description of agents, command queue managers, and coordination queue managers.

Agent queue managers

Normally an MFT agent connects to a single agent queue manager, and uses local queues that are only accessible by that queue manager. The agent is informed which queue manager to connect to, by providing it with the queue manager name when the agent is first created.

With IBM MQ for z/OS, it is possible to create the agent and replace the queue manager name with the name of a queue sharing group (QSG). This means that the agent can connect to any available queue manager in the QSG to perform file transfers. Should there be a failure of the queue manager that the agent is currently connected to, the agent detects the failure and reconnects to an alternative queue manager in the QSG.

Connecting an agent to a QSG in combination with the highly available agent support provided allows very robust MFT topologies to be created. See [“Agents à haute disponibilité dans Managed File Transfer” on page 846](#).

For example, in the following figure *Agent1* has been created so that its agent queue manager is a QSG consisting of two queue managers *QM1* and *QM2*. The agent queues have been defined as shared queues, stored in the coupling facility.

This means that the agent can run on either *LPAR 1* or *LPAR 2* and connect to either *QM1* or *QM2*. The files and data sets that the agent reads from, or writes to, are shared, meaning they can be accessed from either LPAR.

In addition, the agent has been configured to be a highly available agent. In the diagram the agent is active in *LPAR 1* and a standby instance of the agent is running in *LPAR 2*.

This topology provides high resilience. Should either the agent running on *LPAR 1* fail, or queue manager *QM1* fail, or *LPAR 1* fail the standby instance of the agent on *LPAR 2* can take over and carry on processing file transfers from the point of failure.

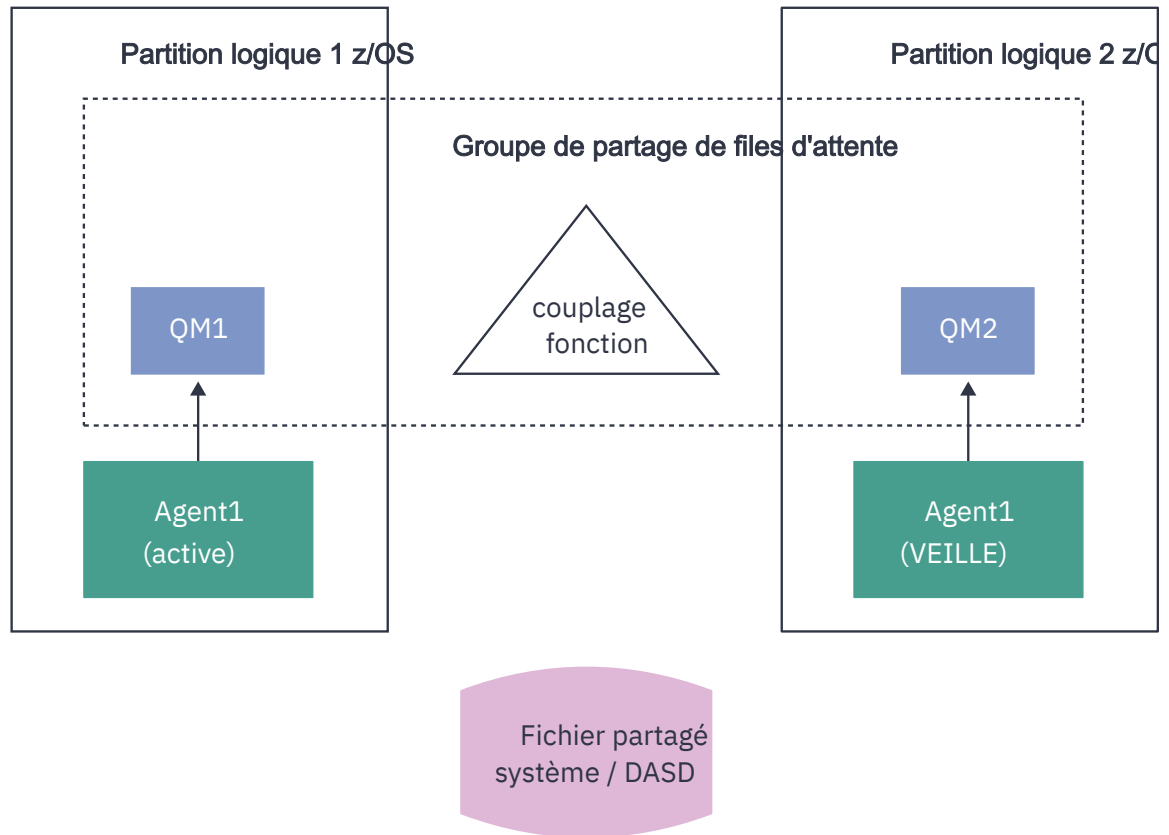


Figure 95. Highly available MFT agent using a queue sharing group

Creating an agent that uses a QSG as the agent queue manager

You create an agent using the `fteCreateAgent` command. When doing this, the name of the queue sharing group is provided for the agent queue manager. For example:

```
fteCreateAgent -agentName Agent1 -agentQMgr QSG1
```

This creates an agent called *Agent1* which uses any queue manager that is a member of QSG *QSG1* as its agent queue manager. In this configuration the agent connects to the agent queue manager using a cross memory (bindings mode) connection which means that the agent and the queue manager must be on the same LPAR. This is exactly like the example shown in figure 1 above.

When you run the **fteCreateAgent** command it generates a set of MQSC commands to create the necessary queues on the agent queue manager.

When the agent queue manager is a QSG, this set of commands needs to be modified so that each queue is created as a shared queue. That is, each queue needs to be created with `QSGDISP(SHARED)` and an appropriate coupling facility structure provided by the `CFSTRUCT` attribute.

The following example shows you how to change the MQSC command for creating the `SYSTEM.FTE.COMMAND.AGENT1` queue as a shared queue. The changes to the defaults are in bold text.

Important: You need to make similar changes to all the other queues that the agent uses.

```
DEFINE QLOCAL(SYSTEM.FTE.COMMAND.AGENT1) +
  QSGDISP(SHARED) +
  CFSTRUCT(MFTSTRUCT) +
  DEFPRTY(0) +
  DEFSOPT(SHARED) +
  GET(ENABLED) +
  INDXTYPE(CORRELID) +
  MAXDEPTH(5000) +
  MAXMSGL(4194304) +
  MSGDLVSQ(PRIORITY) +
  PUT(ENABLED) +
  RETINTVL(99999999) +
  SHARE +
  NOTRIGGER +
  USAGE(NORMAL) +
  REPLACE
```

Creating an agent that uses a QSG as the agent queue manager and connects as a client

Agents can connect to their agent queue manager using a client channel. You can use this approach to allow the agent to run on distributed platforms while connecting to a QSG. If all queue managers in the QSG are licensed for IBM MQ Advanced for z/OS Value Unit Edition, then the agent can also connect to them from a z/OS LPAR that does not have a local queue manager.

This topology is shown in the following figure and allows the agent to take advantage of the resiliency of QSGs. If the queue manager in the QSG that the agent is currently connected to fails, then the agent automatically reconnects to a different member of the QSG and carries on processing.

The sysplex distributor is used to spread the connections from the agent across the available queue managers in the QSG.

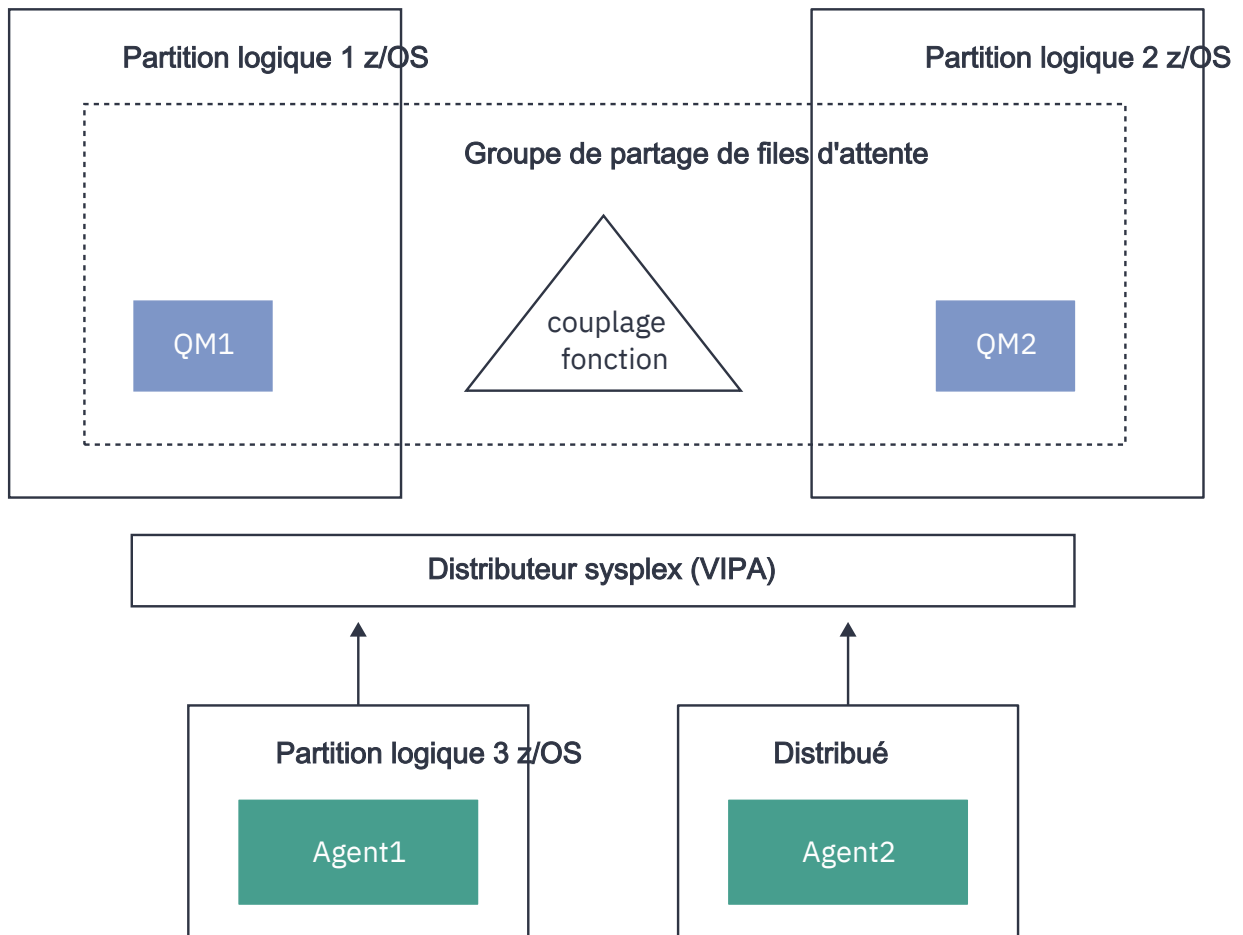


Figure 96. MFT agents connecting to a queue sharing group as a client

In order to make use of this topology, the queue managers in the QSG must each have a server connection channel defined for use by the agent. See [“Connecting a client to a queue sharing group”](#) on page 66 for information on how to do this.

When creating the agent the queue managers need to be configured, so that they can use the channel that is defined to the QSG, and access it through the sysplex distributor. For example:

```
fteCreateAgent -agentName Agent1 -agentQMgr QSG1 -agentQMgrHost vipaAddress
-agentQMgrPort sharedPort -agentQMgrChannel CHANNEL1
```

As previously stated the MQSC commands generated by running the **fteCreateAgent** command must be adjusted to specify QSGDISP(SHARED) and an appropriate coupling facility structure in the CFSTRUCT attribute.

Command queue managers

The MFT command queue manager can be part of a QSG. However, the name of a QSG cannot be used when specifying a command queue manager; you must use a specific queue manager name.

Coordination queue managers

The MFT coordination queue manager can be part of a QSG. However, as with a command queue manager, the name of a QSG cannot be used when specifying a coordination queue manager; you must use a specific queue manager name.

Commands connecting to a QSG

MFT provides a number of commands to managed agents, transfers and agent, command, or coordination queue managers. You can use only those commands that connect to an agent queue manager, if the queue manager is in a QSG.

Following is a list of the commands that connect to the agent queue manager:

- **fteCleanAgent**
- **fteCreateAgent**
- **fteCreateBridgeAgent**
- **fteCreateCDAgent**
- **fteDeleteAgent**

Note that you must provide the name of the queue manager when running other MFT commands.

Using Managed File Transfer for z/OS with the JZOS Java launcher

You can apply the instructions in this topic as an alternative method of using Managed File Transfer in your enterprise, on your IBM MQ for z/OS system.

Overview

Managed File Transfer for z/OS (MFT) uses the standard z/OS installation procedure. An alternative way of running MFT commands is to use JCL and the JZOS Java Launcher.

See [JZOS Batch Launcher and Toolkit](#) for further details.

If your JCL fails to process correctly, see [Common MFT problems with JZOS](#).

Example JCL

```
//JOHNDOEA JOB 1,MSGCLASS=H
// JCLLIB ORDER=(SCEN.MFT.JCL) (1)
// INCLUDE MEMBER=BFGJCL8 (2)
// DD * (2A)
. ${BFG_PROD}/bin/fteBatch createAgent (3)
export IBM_JAVA_OPTIONS="${BFG_JAVA_OPTIONS} ${BFG_LANG}" (4)
export JZOS_MAIN_ARGS="${BFG_MAIN_ARGS}" (4)
//MAINARGS DD *
-agentName MYAGENT (5)
-f
-agentQMgr MQPD
-p MQPD
/*
```

where:

- (1) Is the location of included JCL statements
- (2) Include the specified JCL member from the location in 1)
- (2A) This extends the //STDENV - see below
- (3) This is the command to be executed, without the leading fte prefix
- (4) These lines are required, they set up information for JZOS
- (5) The parameters to the command
- The BFGJCL8 member (you can select your own name) invokes JZOS. This member has the STEPLIB and other JCL needed to run MFT.

Other JCL you need to include

You should include JCL for the IBM MQ for z/OS libraries, and if you are using the Db2 logger, the Db2 libraries.

For example:

```
//WMQFTE EXEC PGM=JVMLDM86,REGION=0M PARM='+T' (1)
//STÉPLIB DD DSN=SYS1.SIEALNKE,DISP=SHR (2)
//* MQ libraries
// DD DSN=MQM.V920.SCSQAUTH,DISP=SHR MQ Bindings
// DD DSN=MQM.V920.SCSQANLE,DISP=SHR MQ Bindings
// DD DSN=MQM.V920.SCSQLOAD,DISP=SHR MQ Bindings

//* DB2 libraries
// DD DISP=SHR,DSN=SYS2.DB2.V12.SDSNEXIT.DBCP
// DD DISP=SHR,DSN=SYS2.DB2.V12.SDSNLOAD
// DD DISP=SHR,DSN=SYS2.DB2.V12.SDSNLOAD2
//SYSOUT DD SYSOUT=H
//SYSPRINT DD SYSOUT=H
//STDOUT DD SYSOUT=H
//STDERR DD SYSOUT=H

//STDENV DD DSN=SCEN.MFT.JCL(BFGZENV8),DISP=SHR (3)
```

where:

- (1) Is the name of the JZOS program. Look in SYS1.SIEALNKE for the version on your system. Add ,PARM='+T' to give additional diagnostics.
- (2) This is the data set with the JZOS program.
- (3) This is the member name of a shell script. It defines parameters needed by MFT. See [“Shell script to define MFT”](#) on page 827.

It can be any data set and member. It needs to be last in the file because the JCL job extends this. See 2A in [“Example JCL”](#) on page 826.

Shell script to define MFT

In the [“Other JCL you need to include”](#) on page 826 example, the member BFGZENV8 is used. This is based on the JZOS profile.

You need to know:

- The location where Java is installed
- The location of the IBM MQ for z/OS Java libraries and the MFT libraries.
- A user ID needs to be in a specific group to be considered as an IBM MQ for z/OS administrator. You need the name of this group
- If you are not using English for the messages, you need to know which language to specify.

Example file

```
# This is a shell script that configures
# any environment variables for the Java JVM.
# Variables must be exported to be seen by the launcher.
# Use PARM='+T' and set -x to debug environment script problems
set -x
# . /etc/profile
#
# Java configuration (including MQ Java interface)
#
export _BPXK_AUTOCVT="ON"
export JAVA_HOME="/java/java71_bit64_sr3_fp30/J7.1_64/"
export PATH="/bin:${JAVA_HOME}/bin/classic/"
LIBPATH="/lib:/usr/lib:${JAVA_HOME}/bin"
LIBPATH="$LIBPATH:${JAVA_HOME}/bin/classic"
LIBPATH=$LIBPATH:"/mqm/V9R2M0/java/lib/"
export LIBPATH

export BFG_JAVA_HOME="${JAVA_HOME}"
export BFG_WTO="YES"
export BFG_GROUP_NAME=MQADM
export BFG_PROD="/mqm/V9R2M0/mqft"
export BFG_CONFIG="/u/johndoe/fteconfig"
```

```
# export BFG_LANG=" -Duser.language=de "  
export BFG_LANG=" "
```

where:

export _BPXK_AUTOCVT="ON"

Is required for Unicode conversion

export JAVA_HOME="/java/java71_bit64/J7.1_64/"

Is the location of the Java directory. Specify the name of the path for Java. This directory contains bin and other directories.

export PATH="/bin:\${JAVA_HOME}/bin/classic/"

Sets up the path statement for Java executable statements

LIBPATH="/lib:/usr/lib:\${JAVA_HOME}/bin"

Sets up the library path for the Java executable statements

LIBPATH="\$LIBPATH:\${JAVA_HOME}/bin/classic"

Adds more Java libraries to the LIBPATH statement.

LIBPATH=\$LIBPATH:"/mqm/V9R2M0/java/lib/"

Adds IBM MQ for z/OS libraries in the library path. Specify the name of your IBM MQ for z/OS libraries in z/OS UNIX System Services.

export LIBPATH

Makes the LIBPATH available to JZOS

export BFG_JAVA_HOME="\${JAVA_HOME}"

Sets the BFG_JAVA_HOME to the value of JAVA_HOME specified above

export BFG_WTO="YES"

Setting BFG_WTO to YES causes messages to be displayed on the joblog using WTO

export BFG_GROUP_NAME=MQADM

User IDs, which are a member of the specified group, are considered IBM MQ for z/OS administrators

export BFG_PROD="/mqm/V9R2M0/mqft"

Is the path where the MFT code is located

export BFG_DATA="/u/johndoe/fteconfig"

Is where the MFT configuration information is stored

export BFG_LANG=" -Duser.language=de "

Is a commented out statement to define the language as German

export BFG_LANG=" "

Specifies the language as the default, English.

The contents of the MFT product in `/lib/messages/BFGNVMessages_*.properties` lists the languages available. The default is to leave the value blank, which means that English is used.

Related tasks

[“Configuring Managed File Transfer for z/OS” on page 796](#)

[Managed File Transfer for z/OS requires customization to enable the component to operate correctly.](#)

[Planning for Managed File Transfer](#)

IBM i Configuration de MFT sur IBM i

Pour commencer à utiliser Managed File Transfer après l'avoir installé, vous devez effectuer une configuration pour votre gestionnaire de files d'attente de coordination et votre agent.

Pourquoi et quand exécuter cette tâche

Une fois l'installation terminée, vous devez exécuter les scripts de configuration fournis par Managed File Transfer pour les nouveaux gestionnaires de files d'attente de coordination et les nouveaux agents avant

de pouvoir utiliser les gestionnaires de files d'attente de coordination et les agents pour transférer des fichiers. Vous devez ensuite démarrer les agents que vous avez créés.

Procédure

1. Pour tous les nouveaux gestionnaires de files d'attente de coordination: exécutez les commandes MQSC dans le fichier *coordination_qmgr_name.mqsc* sur le gestionnaire de files d'attente de coordination. Si le gestionnaire de files d'attente de coordination ne se trouve pas sur le même ordinateur que l'installation, copiez le fichier script MQSC sur l'ordinateur où se trouve le gestionnaire de files d'attente, puis exécutez le script.
 - a) A partir d'une ligne de commande IBM i , démarrez qshell à l'aide de la commande suivante: CALL QSHELL
 - b) Accédez au répertoire suivant : /QIBM/UserData/mqm/mqft/config/*coordination_qmgr_name*
 - c) Exécutez la commande suivante en remplaçant *coordination_qmgr_name* par le nom de votre gestionnaire de files d'attente:

```
/QSYS.LIB/QMQM.LIB/RUNMQSC.PGM coordination_qmgr_name < coordination_qmgr_name.mqsc
```

Vous pouvez configurer le gestionnaire de files d'attente de coordination manuellement à la place. Pour plus d'informations, voir [«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 833.

2. Pour tous les nouveaux agents: exécutez les commandes MQSC dans le fichier *agent_name_create.mqsc* sur le gestionnaire de files d'attente d'agent.

Si le gestionnaire de files d'attente d'agent ne se trouve pas sur le même ordinateur que l'agent, copiez le fichier script MQSC sur l'ordinateur où se trouve le gestionnaire de files d'attente, puis exécutez le script.

 - a) A partir d'une ligne de commande IBM i , démarrez qshell à l'aide de la commande suivante: CALL QSHELL
 - b) Accédez au répertoire suivant : /QIBM/UserData/mqm/mqft/config/*agent_qmgr_name*/agents
 - c) Exécutez la commande suivante en remplaçant *agent_qmgr_name* par le nom de votre gestionnaire de files d'attente d'agent et en remplaçant *agent_name* par le nom de votre agent:

```
/QSYS.LIB/QMQM.LIB/RUNMQSC.PGM agent_qmgr_name < agent_name_create.mqsc
```

Vous pouvez configurer le gestionnaire de files d'attente d'agent manuellement à la place. Pour plus d'informations, voir [«Configuration des gestionnaires de files d'attente d'agent MFT»](#), à la page 841.

3. Si vous n'avez pas encore démarré le sous-système QMFT dans le cadre de l'installation, à partir de la ligne de commande IBM i , démarrez le sous-système QMFT à l'aide de la commande suivante: STRSBS SBSD(QMQMMFT/QMFT) ou STRSBS QMQMMFT/QMFT
4. Démarrez vos nouveaux agents à l'aide de la commande **fteStartAgent** .
 - a) A partir d'une ligne de commande IBM i , démarrez qshell à l'aide de la commande suivante: CALL QSHELL
 - b) Accédez au répertoire suivant : /QIBM/ProdData/mqm/bin
 - c) Exécutez la commande suivante, en remplaçant AGENT par le nom de votre agent:

```
./fteStartAgent AGENT
```

Que faire ensuite

Il est recommandé de configurer des bacs à sable pour limiter les zones du système de fichiers auxquelles un agent peut accéder. Cette fonction est décrite dans [Utilisation des bacs à sable d'agent MFT](#).

Concepts associés

[«Configuration de MFT pour la première utilisation»](#), à la page 830

Vous devez effectuer certaines tâches de configuration pour les agents et les gestionnaires de files d'attente Managed File Transfer une seule fois, la première fois que vous souhaitez les utiliser.

Configuration de MFT pour la première utilisation

Vous devez effectuer certaines tâches de configuration pour les agents et les gestionnaires de files d'attente Managed File Transfer une seule fois, la première fois que vous souhaitez les utiliser.

Concepts associés

[«connexion à IBM MQ»](#), à la page 831

Toutes les communications réseau avec les gestionnaires de files d'attente IBM MQ, y compris les communications liées à Managed File Transfer, impliquent des canaux IBM MQ. Un canal IBM MQ représente une extrémité d'une liaison réseau. Les canaux sont classés en tant que canaux de message ou canaux MQI.

[«Configuration d'un gestionnaire de files d'attente multi-instance pour une utilisation avec MFT»](#), à la page 837

IBM WebSphere MQ 7.0.1 et versions ultérieures prennent en charge la création de gestionnaires de files d'attente multi-instance. Un gestionnaire de files d'attente multi-instance redémarre automatiquement sur un serveur de secours. Managed File Transfer prend en charge la connexion à des gestionnaires de files d'attente d'agent multi-instance, à un gestionnaire de files d'attente de coordination multi-instance et à un gestionnaire de files d'attente de commandes multi-instance.

Tâches associées

[«Configuration des gestionnaires de files d'attente réseau MFT»](#), à la page 832

Si votre réseau Managed File Transfer inclut plusieurs gestionnaires de files d'attente IBM MQ, ces gestionnaires de files d'attente IBM MQ doivent pouvoir communiquer à distance entre eux.

[«Configuration des gestionnaires de files d'attente d'agent MFT»](#), à la page 841

Après l'installation, exécutez le script `agent_name_create.mqsc` dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name` pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente d'agent. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez ces étapes sur le gestionnaire de files d'attente d'agent.

[«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 833

Après avoir exécuté la commande **fteSetupCoordination**, exécutez le script `coordination_qmgr_name.mqsc` dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name` pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente de coordination. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez les étapes suivantes sur le gestionnaire de files d'attente de coordination.

[«Creating an MFT Agent or Logger command data set»](#), à la page 795

You can create a PDSE data set of commands from the Managed File Transfer command template data set for a specific Managed File Transfer Agent or Managed File Transfer Logger for a specific coordination.

[«Updating an existing MFT Agent or Logger command data set on z/OS»](#), à la page 807

You can update an Managed File Transfer command PDSE library data set that is created from the Managed File Transfer command template data set.

Référence associée

[Paramètres de file d'attente de l'agent MFT](#)

[Les files d'attente système MFT et la rubrique système](#)

[«Conservation des messages de journal MFT»](#), à la page 839

Managed File Transfer envoie la progression du transfert de fichiers et les informations de journal au gestionnaire de files d'attente de coordination. Le gestionnaire de file d'attente de coordination publie ces

informations dans tous les abonnements correspondants à SYSTEM.FTE . S'il n'y a pas d'abonnement, ces informations ne sont pas conservées.

connexion à IBM MQ

Toutes les communications réseau avec les gestionnaires de files d'attente IBM MQ , y compris les communications liées à Managed File Transfer, impliquent des canaux IBM MQ . Un canal IBM MQ représente une extrémité d'une liaison réseau. Les canaux sont classés en tant que canaux de message ou canaux MQI.

Managed File Transfer et canaux

Managed File Transfer utilise des canaux MQI pour connecter des agents en mode client à leurs gestionnaires de files d'attente d'agent et pour connecter des applications de commande (par exemple, **fteCreateTransfer**) à leurs gestionnaires de files d'attente de commande et de coordination. Dans la configuration par défaut, ces connexions sont établies à l'aide d'un canal SVRCONN appelé SYSTEM.DEF.SVRCONN, qui existe par défaut sur tous les gestionnaires de files d'attente. En raison de ces valeurs par défaut, vous n'avez pas besoin de modifier les canaux MQI pour une installation Managed File Transfer de base.

Il existe six types de points de terminaison de canal de message, mais cette rubrique ne couvre que les paires émetteur-récepteur. Pour plus d'informations sur les autres combinaisons de canaux, voir [Composants de la mise en file d'attente répartie](#) .

Chemins d'accès aux messages requis

Les messages IBM MQ ne pouvant circuler que sur les canaux de transmission de messages, vous devez vous assurer que les canaux sont disponibles pour tous les chemins de messages requis par Managed File Transfer. Ces chemins n'ont pas besoin d'être directs ; les messages peuvent transiter par des gestionnaires de files d'attente intermédiaires si nécessaire. Cette rubrique couvre uniquement la communication directe point à point. Pour plus d'informations sur ces options, voir [Comment accéder au gestionnaire de files d'attente éloignées](#) .

Les chemins de communication utilisés par Managed File Transfer sont les suivants:

Agent à agent

Les deux agents entre lesquels les fichiers sont transférés requièrent une communication bidirectionnelle entre leurs gestionnaires de files d'attente associés. Etant donné que ce chemin contient les données non formatées, pensez à le rendre aussi court, rapide ou bon marché que possible en fonction de vos besoins.

Agent à la coordination

Les messages de journal provenant des agents qui participent à un transfert doivent pouvoir atteindre le gestionnaire de files d'attente de coordination.

Commande à l'agent

Tout gestionnaire de files d'attente auquel se connectent des applications de commande ou IBM MQ Explorer (à l'aide du gestionnaire de files d'attente de commandes) doit pouvoir envoyer des messages aux gestionnaires de files d'attente des agents que ces applications de commande sont utilisées pour contrôler. Pour que les messages de retour soient affichés par les commandes, utilisez une connexion bidirectionnelle.

Pour plus d'informations, voir *Vérification d'une installation IBM MQ* pour la ou les plateformes utilisées par votre entreprise.

Concepts associés

[«Configuration d'un gestionnaire de files d'attente multi-instance pour une utilisation avec MFT», à la page 837](#)

IBM WebSphere MQ 7.0.1 et versions ultérieures prennent en charge la création de gestionnaires de files d'attente multi-instance. Un gestionnaire de files d'attente multi-instance redémarre automatiquement sur un serveur de secours. Managed File Transfer prend en charge la connexion à des gestionnaires de

files d'attente d'agent multi-instance, à un gestionnaire de files d'attente de coordination multi-instance et à un gestionnaire de files d'attente de commandes multi-instance.

Tâches associées

«[Configuration des gestionnaires de files d'attente réseau MFT](#)», à la page 832

Si votre réseau Managed File Transfer inclut plusieurs gestionnaires de files d'attente IBM MQ , ces gestionnaires de files d'attente IBM MQ doivent pouvoir communiquer à distance entre eux.

«[Configuration du gestionnaire de files d'attente de coordination pour MFT](#)», à la page 833

Après avoir exécuté la commande **fteSetupCoordination** , exécutez le script `coordination_qmgr_name.mqsc` dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name` pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente de coordination. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez les étapes suivantes sur le gestionnaire de files d'attente de coordination.

Configuration des gestionnaires de files d'attente réseau MFT

Si votre réseau Managed File Transfer inclut plusieurs gestionnaires de files d'attente IBM MQ , ces gestionnaires de files d'attente IBM MQ doivent pouvoir communiquer à distance entre eux.

Pourquoi et quand exécuter cette tâche

Vous pouvez configurer vos gestionnaires de files d'attente de deux façons pour qu'ils puissent communiquer les uns avec les autres :

- En configurant un cluster de gestionnaires de files d'attente IBM MQ.

Pour plus d'informations sur les clusters de gestionnaires de files d'attente IBM MQ et sur leur configuration, voir «[Configuration d'un cluster de gestionnaires de files d'attente](#)», à la page 316.

- En configurant des canaux entre les gestionnaires de files d'attente, comme suit :

Configuration de canaux entre des gestionnaires de files d'attente

Configurez les canaux de transmission de messages suivants entre vos gestionnaires de files d'attente :

- Depuis le gestionnaire de file d'attente d'agent vers le gestionnaire de file d'attente de coordination
- Depuis le gestionnaire de file d'attente de commandes vers le gestionnaire de file d'attente d'agent
- Depuis le gestionnaire de file d'attente d'agent vers le gestionnaire de file d'attente de commandes (pour activer l'affichage des messages de retour par les commandes)
- Depuis le gestionnaire de file d'attente de commandes vers le gestionnaire de file d'attente de coordination
- Du gestionnaire de files d'attente d'agent vers tout autre gestionnaire de files d'attente d'agent du réseau Managed File Transfer

Pour plus d'informations sur la configuration de cette communication, commencez par lire la rubrique relative à [l'administration des objets IBM MQ distants avec MQSC](#).

Voici quelques exemples d'étapes suggérées :

Procédure

1. Créez dans le gestionnaire de files d'attente IBM MQ une file d'attente de transmission du même nom que le gestionnaire de file d'attente de coordination.

Vous pouvez utiliser la commande MQSC suivante :

```
DEFINE QLOCAL(coordination-qmgr-name) USAGE(XMITQ)
```

2. Dans le gestionnaire de files d'attente IBM MQ, créez un canal émetteur vers le gestionnaire de file d'attente de coordination Managed File Transfer.

Le nom de la file d'attente de transmission créée à l'étape précédente est un paramètre requis pour ce canal.

Pour les agents sous Managed File Transfer for IBM MQ, les messages sont publiés avec un format vide.

Vous pouvez utiliser la commande MQSC suivante :

```
DEFINE CHANNEL(channel-name) CHLTYPE(SDR) CONNAME('coordination-qmgr-host(coordination-qmgr-port)')
XMITQ(coordination-qmgr-name) CONVERT(NO)
```

Remarque : Définissez CONVERT(NO) uniquement si nécessaire.

3. Dans le gestionnaire de file d'attente de coordination Managed File Transfer, créez un canal récepteur vers le gestionnaire de files d'attente IBM MQ. Attribuez-lui le nom du canal émetteur dans le gestionnaire de files d'attente IBM MQ.

Vous pouvez utiliser la commande MQSC suivante :

```
DEFINE CHANNEL(channel-name) CHLTYPE(RCVR)
```

Que faire ensuite

Suivez ensuite les étapes de configuration de votre gestionnaire de files d'attente de coordination: [«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 833.

Concepts associés

[«connexion à IBM MQ»](#), à la page 831

Toutes les communications réseau avec les gestionnaires de files d'attente IBM MQ, y compris les communications liées à Managed File Transfer, impliquent des canaux IBM MQ. Un canal IBM MQ représente une extrémité d'une liaison réseau. Les canaux sont classés en tant que canaux de message ou canaux MQI.

[«Configuration d'un gestionnaire de files d'attente multi-instance pour une utilisation avec MFT»](#), à la page 837

IBM WebSphere MQ 7.0.1 et versions ultérieures prennent en charge la création de gestionnaires de files d'attente multi-instance. Un gestionnaire de files d'attente multi-instance redémarre automatiquement sur un serveur de secours. Managed File Transfer prend en charge la connexion à des gestionnaires de files d'attente d'agent multi-instance, à un gestionnaire de files d'attente de coordination multi-instance et à un gestionnaire de files d'attente de commandes multi-instance.

Tâches associées

[«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 833

Après avoir exécuté la commande **fteSetupCoordination**, exécutez le script *coordination_qmgr_name.mqsc* dans le répertoire *MQ_DATA_PATH/mqft/config/coordination_qmgr_name* pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente de coordination. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez les étapes suivantes sur le gestionnaire de files d'attente de coordination.

Configuration du gestionnaire de files d'attente de coordination pour MFT

Après avoir exécuté la commande **fteSetupCoordination**, exécutez le script *coordination_qmgr_name.mqsc* dans le répertoire *MQ_DATA_PATH/mqft/config/coordination_qmgr_name* pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente de coordination. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez les étapes suivantes sur le gestionnaire de files d'attente de coordination.

Pourquoi et quand exécuter cette tâche

Procédure

1. Créez une file d'attente locale nommée SYSTEM.FTE.
2. Ajoutez le système SYSTEM.FTE , file d'attente vers SYSTEM.QPUBSUB.QUEUE.NAMELIST .
3. Créez une rubrique nommée SYSTEM.FTE avec une chaîne de rubrique SYSTEM.FTE.
4. Vérifiez les attributs de distribution de messages non persistants (NPMSGDLV) et de distribution de messages persistants (PMSGDLV) de SYSTEM.FTE est définie sur ALLAVAIL.
5. Vérifiez que l'attribut de mode de publication / abonnement (PSMODE) du gestionnaire de files d'attente de coordination est défini sur ENABLED.

Que faire ensuite

Si vous exécutez la commande `strmqm -c` sur un gestionnaire de files d'attente qui a été configuré en tant que gestionnaire de files d'attente de coordination, la commande supprime la modification apportée à l' [étape 2](#) (ajout de SYSTEM.FTE , file d'attente vers SYSTEM.QPUBSUB.QUEUE.NAMELIST liste de noms). En effet, `strmqm -c` recrée les objets IBM MQ par défaut et inverse les modifications Managed File Transfer . Par conséquent, si vous avez démarré le gestionnaire de files d'attente avec `strmqm -c`, effectuez l'une des étapes suivantes:

- Exécutez à nouveau le script `coordination_qmgr_name.mqsc` sur le gestionnaire de files d'attente.
- Répétez l' [étape 2](#).

Concepts associés

[«connexion à IBM MQ»](#), à la [page 831](#)

Toutes les communications réseau avec les gestionnaires de files d'attente IBM MQ , y compris les communications liées à Managed File Transfer, impliquent des canaux IBM MQ . Un canal IBM MQ représente une extrémité d'une liaison réseau. Les canaux sont classés en tant que canaux de message ou canaux MQI.

[«Configuration d'un gestionnaire de files d'attente multi-instance pour une utilisation avec MFT»](#), à la [page 837](#)

IBM WebSphere MQ 7.0.1 et versions ultérieures prennent en charge la création de gestionnaires de files d'attente multi-instance. Un gestionnaire de files d'attente multi-instance redémarre automatiquement sur un serveur de secours. Managed File Transfer prend en charge la connexion à des gestionnaires de files d'attente d'agent multi-instance, à un gestionnaire de files d'attente de coordination multi-instance et à un gestionnaire de files d'attente de commandes multi-instance.

Tâches associées

[«Configuration des gestionnaires de files d'attente réseau MFT»](#), à la [page 832](#)

Si votre réseau Managed File Transfer inclut plusieurs gestionnaires de files d'attente IBM MQ , ces gestionnaires de files d'attente IBM MQ doivent pouvoir communiquer à distance entre eux.

Référence associée

[fteSetupCoordination](#)

Création d'une structure de transfert de fichiers IBM MQ

Vous pouvez configurer une structure Managed File Transfer , basée sur un agent unique connecté à un gestionnaire de files d'attente sur la même machine.

Pourquoi et quand exécuter cette tâche

La configuration MFT est stockée dans une structure de fichiers sous IBM MQ DataPath, sur la machine sur laquelle l'agent sera localisé.

L'exemple de configuration suivant concerne un gestionnaire de files d'attente MFT in IBM MQ 8.0 nommé SAMPLECOORD (avec la sécurité désactivée) et un agent MFT unique nommé SAMPLEAGENT:

```
+--- config
+--- SAMPLECOORD
```

```

+--- command.properties
+--- coordination.properties
+--- SAMPLECOORD.mqsc
+--- agents
      +--- SAMPLEAGENT
            +--- agent.properties
            +--- SAMPLEAGENT_create.mqsc
            +--- SAMPLEAGENT_delete.mqsc

+---logs
      +--- SAMPLECOORD
            +--- agents
                  +--- SAMPLEAGENT
                        +--- logs

```

Cet exemple suppose que la sécurité du gestionnaire de files d'attente a été désactivée. Les commandes suivantes, exécutées dans **runmqsc**, désactivent la sécurité après le redémarrage du gestionnaire de files d'attente:

```

runmqsc queue manager
alter qmgr CONNAUTH(NONE);
alter qmgr CHLAUTH(DISABLED);
end;

```

Pour la configuration avec la sécurité activée dans MFT dans IBM MQ 8.0 ou version ultérieure, **CONNAUTH** requiert toutes les commandes MFT qui se connectent à un gestionnaire de files d'attente pour fournir les données d'identification par ID utilisateur et mot de passe. Vous pouvez appliquer les paramètres supplémentaires **-mquserid** et **-mqpassword** pour chaque commande ou définir un fichier `MQMFTCredentials.xml`. L'exemple de fichier de données d'identification suivant définit l'ID utilisateur de `fteuser`, pour lequel le mot de passe `MyPassword` doit être utilisé lors de la connexion au gestionnaire de files d'attente `SAMPLECOORD`:


```

<tns:mqmftCredentials xmlns:tns="http://wmqfte.ibm.com/MQMFTCredentials"
  xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://wmqfte.ibm.com/MQMFTCredentials MQMFTCredentials.xsd">
  <tns:qmgr mqPassword="MyPassword" MyUserId="fteuser" name="SAMPLECOORD"/>
</tns:mqmftCredentials>

```

Pour plus d'informations, voir [Authentification des connexions MFT et IBM MQ](#).

Remarques :

- Pour localiser votre répertoire de configuration MFT, utilisez la commande **fteDisplayVersion -v**.
-  Pour les utilisateurs z/OS, le fichier `MQMFTCredential.xml` peut être localisé en tant que membre dans un fichier partitionné avec un format d'enregistrement variable (RECFM = V) ou un format d'enregistrement non défini (RECFM = U).
- Pour une configuration avec la sécurité activée, ajoutez le paramètre suivant aux étapes ci-dessous afin d'associer les données d'identification au gestionnaire de files d'attente approprié: `-F full_credential_file_path`.
- Le mot de passe en texte clair dans `MQMFTCredential.xml` peut être brouillé à l'aide de la commande suivante:

```

fteObfuscate -f full_file_path_to_MQMFTCredentials.xml

```

Procédure

1. Créez un gestionnaire de files d'attente de coordination.

Un gestionnaire de files d'attente de coordination est un gestionnaire de files d'attente unique, utilisé pour recevoir toutes les informations de statut et de journal de transfert de ses agents. Exécutez ensuite la commande suivante :

```
fteSetupCoordination -coordinationQMgr coordination_qmgr_name
```

Cela crée la configuration de niveau supérieur de base et crée un fichier script IBM MQ pour appeler *coordination_qmgr_name.mqsc*.

La configuration doit ensuite être chargée dans le gestionnaire de files d'attente, en exécutant la commande IBM MQ suivante:

```
runmqsc queue manager name < coordination_qmgr_name.mqsc
```

Remarque : Pour la connexion client TCP à un gestionnaire de files d'attente, vous pouvez utiliser:

```
fteSetupCoordination -coordinationQMgr coordination_qmgr_name  
-coordinationQMgrHost coordination_qmgr_host -coordinationQMgrPort coordination_qmgr_port  
-coordinationQMgrChannel coordination_qmgr_channel
```

Pour le *coordination_qmgr_name.mqsc* créé, vous devez exécuter la commande **runmqsc** sur la même machine que le gestionnaire de files d'attente de coordination.

2. Créez le gestionnaire de files d'attente de commandes.

Un gestionnaire de files d'attente de commandes est un gestionnaire de files d'attente unique qui a été préconfiguré de sorte que l'infrastructure IBM MQ puisse acheminer les demandes MFT à l'agent approprié. Exécutez ensuite la commande suivante :

```
fteSetupCommands -connectionQMgr Command QM Name -p Coordination QM Name
```

Un fichier *command.properties* est créé dans le répertoire de coordination. Notez que *-p* est facultatif et n'est pas obligatoire si les commandes sont configurées pour la coordination par défaut.

Remarque : Pour la connexion client TCP à un gestionnaire de files d'attente, vous pouvez utiliser:

```
fteSetupCommands -p coordination_qmgr_name -commandQMgr connection_qmgr_name  
-commandQMgrHost connection_qmgr_host -commandQMgrPort connection_qmgr_port  
-commandQMgrChannel connection_qmgr_channel
```

3. Créez l'agent.

Un agent est une application qui peut envoyer et recevoir des fichiers. Exécutez ensuite la commande suivante :

```
fteCreateAgent -p coordination_qmgr_name -agentName agent_name -agentQMgr agent_qmgr_name
```

Cela crée la configuration de l'agent sous la coordination et crée un fichier script IBM MQ pour appeler *agent_name.mqsc* dans le répertoire de configuration de l'agent.

Exécutez la commande IBM MQ suivante pour charger le fichier script IBM MQ dans le gestionnaire de files d'attente:

```
runmqsc agent_qmgr_name < agent_name_create.mqsc file
```

Remarque : Pour la connexion client TCP à un gestionnaire de files d'attente, vous pouvez utiliser:

```
fteCreateAgent -p coordination_qmgr_name -agentName agent_name -agentQMgr agent_qmgr_name  
-agentQMgrHost agent_qmgr_host -agentQMgrPort agent_qmgr_port -agentQMgrChannel  
agent_qmgr_channel
```

4. Démarrez l'agent.

Exécutez ensuite la commande suivante :

```
fteStartAgent -p coordination_qmgr_name agentName
```

L'agent démarre en arrière-plan et l'invite de commande est renvoyée. Pour vérifier que l'agent est en cours d'exécution, exécutez la commande suivante:

```
fteListAgents -p coordination_qmgr_name
```

Affiche le statut des agents. Si l'agent s'exécute correctement, il est signalé comme étant à l'état PRET.

Résultats

Une infrastructure MFT de base est prête à être utilisée et vous pouvez maintenant utiliser la commande **fteCreateTransfer** pour demander un transfert. Sinon, si le IBM MQ Explorer est disponible, utilisez les plug-in MFT pour créer et surveiller les transferts.

Vous pouvez ajouter d'autres agents à la configuration en répétant l'étape 3: Créer l'agent. Si la connexion client TCP est utilisée, elle peut se trouver sur des machines différentes. Pour les différentes machines, les commandes **fteSetupCoordination** et **fteSetupCommands** doivent être répétées pour chaque machine, mais les scripts mqsc n'ont pas besoin d'être exécutés.

Les configurations plus complexes peuvent comporter des gestionnaires de files d'attente distincts pour la coordination et chaque agent. Dans ces cas, les différents gestionnaires de files d'attente devront être connectés ensemble.

Concepts associés

Que faire si votre agent MFT n'est pas répertorié par la commande **fteListAgents** ?

Référence associée

[fteSetupCoordination](#)

Commandes fteSetup: créez le fichier MFT [command.properties](#)

[fteCreateAgent](#)

fteObfuscate: [chiffrement des données sensibles](#)

[Format du fichier de données d'identification MFT](#)

[Le fichier MFTagent.properties](#)

Configuration d'un gestionnaire de files d'attente multi-instance pour une utilisation avec MFT

IBM WebSphere MQ 7.0.1 et versions ultérieures prennent en charge la création de gestionnaires de files d'attente multi-instance. Un gestionnaire de files d'attente multi-instance redémarre automatiquement sur un serveur de secours. Managed File Transfer prend en charge la connexion à des gestionnaires de files d'attente d'agent multi-instance, à un gestionnaire de files d'attente de coordination multi-instance et à un gestionnaire de files d'attente de commandes multi-instance.

Configuration d'un gestionnaire de files d'attente multi-instance

Important : Pour plus d'informations sur la configuration d'un gestionnaire de files d'attente multi-instance IBM MQ , voir «[Gestionnaires de files d'attente multi-instance](#)», à la page 547. Veillez à lire ces informations avant de tenter de configurer un gestionnaire de files d'attente multi-instance pour qu'il fonctionne avec Managed File Transfer.

Utilisation d'un gestionnaire de files d'attente multi-instance en tant que gestionnaire de files d'attente d'agent

Pour permettre à un agent de se connecter à la fois à l'instance active et à l'instance de secours de votre gestionnaire de files d'attente multi-instance, ajoutez la propriété `agentQMgrStandby` au fichier `agent.properties` de l'agent. La propriété `agentQMgrStandby` définit le nom d'hôte et le numéro de port utilisés pour les connexions client pour l'instance de gestionnaire de files d'attente

de secours. La valeur de la propriété doit être indiquée au format MQ CONNAME, c'est-à-dire `host_name(port_number)`.

La propriété `agentQMgr` indique le nom du gestionnaire de files d'attente multi-instance. La propriété `agentQMgrHost` indique le nom d'hôte de l'instance de gestionnaire de files d'attente active et la propriété `agentQMgrPort` indique le numéro de port de l'instance de gestionnaire de files d'attente active. L'agent doit se connecter en mode client à l'instance active et à l'instance de secours du gestionnaire de files d'attente multi-instance.

Pour plus d'informations, voir [Le fichier MFT agent.properties](#).

Cet exemple présente le contenu du fichier `agent.properties` pour AGENT1 qui se connecte à un gestionnaire de files d'attente multi-instance appelé QM_JUPITER. L'instance active de QM_JUPITER se trouve sur le système `host1` et utilise le numéro de port 1414 pour les connexions client. L'instance de secours de QM_JUPITER se trouve sur le système `host2` et utilise le numéro de port 1414 pour les connexions client.

```
agentName=AGENT1
agentDesc=
agentQMgr=QM_JUPITER
agentQMgrPort=1414
agentQMgrHost=host1
agentQMgrChannel=SYSTEM.DEF.SVRCONN
agentQMgrStandby=host2(1414)
```

Utilisation d'un gestionnaire de files d'attente multi-instance comme gestionnaire de files d'attente de coordination

Pour activer les connexions à l'instance active et à l'instance de secours de votre gestionnaire de files d'attente de coordination multi-instance, ajoutez la propriété `coordinationQMgrStandby` à tous les fichiers `coordination.properties` de votre topologie Managed File Transfer.

Pour plus d'informations, voir [Le fichier MFT coordination.properties](#).

Cet exemple illustre le contenu d'un fichier `coordination.properties` qui spécifie les détails de connexion à un gestionnaire de files d'attente de coordination multi-instance appelé QM_SATURN. L'instance active de QM_SATURN se trouve sur le système `coordination_host1` et utilise le numéro de port 1420 pour les connexions client. L'instance de secours de QM_SATURN se trouve sur le système `coordination_host2` et utilise le numéro de port 1420 pour les connexions client.

```
coordinationQMgr=QM_SATURN
coordinationQMgrHost=coordination_host1
coordinationQMgrPort=1420
coordinationQMgrChannel=SYSTEM.DEF.SVRCONN
coordinationQMgrStandby=coordination_host2(1420)
```

Le consignateur autonome Managed File Transfer doit toujours se connecter à son gestionnaire de files d'attente en mode liaisons. Lorsque vous utilisez le consignateur autonome avec un gestionnaire de files d'attente de coordination multi-instance, connectez le consignateur autonome, en mode liaisons, à un gestionnaire de files d'attente différent. Les étapes à suivre sont décrites dans «Autres configurations pour un consignateur MFT autonome», à la page 865. Vous devez définir les canaux entre le gestionnaire de files d'attente du consignateur autonome et le gestionnaire de files d'attente de coordination avec le nom d'hôte et le numéro de port des deux instances du gestionnaire de files d'attente de coordination multi-instance. Pour plus d'informations sur la procédure à suivre, voir «Gestionnaires de files d'attente multi-instance», à la page 547.

Le plug-in Managed File Transfer pour IBM MQ Explorer se connecte au gestionnaire de files d'attente de coordination en mode client. Si l'instance active du gestionnaire de files d'attente de coordination multi-instance échoue, l'instance de secours du gestionnaire de files d'attente de coordination devient active et le plug-in se reconnecte.

Les commandes Managed File Transfer **fteList*** et **fteShowAgentDetails** se connectent directement au gestionnaire de file d'attente de coordination. Si l'instance active de la coordination multi-instance n'est pas disponible, ces commandes tentent de se connecter à l'instance de secours du gestionnaire de files d'attente de coordination.

Utilisation d'un gestionnaire de files d'attente multi-instance comme gestionnaire de files d'attente de commandes

Pour activer les connexions à l'instance active et à l'instance de secours de votre gestionnaire de files d'attente de commandes multi-instance, ajoutez la propriété `connectionQMgrStandby` à tous les fichiers `command.properties` de votre topologie Managed File Transfer .

Voir [Le fichier MFT command.properties](#) pour plus d'informations.

Cet exemple illustre le contenu d'un fichier `command.properties` qui spécifie les détails de connexion à un gestionnaire de files d'attente de commandes multi-instance appelé QM_MARS. L'instance active de QM_MARS se trouve sur le système `command_host1` et utilise le numéro de port 1424 pour les connexions client. L'instance de secours de QM_MARS se trouve sur le système `command_host2` et utilise le numéro de port 1424 pour les connexions client.

```
connectionQMgr=QM_SATURN
connectionQMgrHost=command_host1
connectionQMgrPort=1424
connectionQMgrChannel=SYSTEM.DEF.SVRCONN
connectionQMgrStandby=command_host2(1424)
```

Concepts associés

[«connexion à IBM MQ»](#), à la page 831

Toutes les communications réseau avec les gestionnaires de files d'attente IBM MQ , y compris les communications liées à Managed File Transfer, impliquent des canaux IBM MQ . Un canal IBM MQ représente une extrémité d'une liaison réseau. Les canaux sont classés en tant que canaux de message ou canaux MQI.

Tâches associées

[«Configuration des gestionnaires de files d'attente réseau MFT»](#), à la page 832

Si votre réseau Managed File Transfer inclut plusieurs gestionnaires de files d'attente IBM MQ , ces gestionnaires de files d'attente IBM MQ doivent pouvoir communiquer à distance entre eux.

[«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 833

Après avoir exécuté la commande **fteSetupCoordination** , exécutez le script `coordination_qmgr_name.mqsc` dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name` pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente de coordination. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez les étapes suivantes sur le gestionnaire de files d'attente de coordination.

Conservation des messages de journal MFT

Managed File Transfer envoie la progression du transfert de fichiers et les informations de journal au gestionnaire de files d'attente de coordination. Le gestionnaire de file d'attente de coordination publie ces informations dans tous les abonnements correspondants à `SYSTEM.FTE` . S'il n'y a pas d'abonnement, ces informations ne sont pas conservées.

Moyens de garantir la conservation des informations

Si la progression du transfert ou les informations de journal sont importantes pour votre entreprise, vous devez prendre l'une des mesures suivantes pour vous assurer que les informations sont conservées:

- Utilisez le consignateur de base de données Managed File Transfer pour copier les messages publiés dans `SYSTEM.FTE/Log` dans une base de données Oracle ou Db2 .

- Définissez un abonnement à SYSTEM.FTE , qui stocke les publications dans une file d'attente IBM MQ . Définissez cet abonnement avant de transférer des transferts de fichiers pour vous assurer que tous les messages de progression et de journal sont conservés dans la file d'attente.
- Ecrivez une application qui utilise l'interface de file d'attente de messages (MQI) ou IBM MQ JMS pour créer un abonnement durable et traiter les publications qui sont distribuées à l'abonnement. Cette application doit être opérationnelle avant tout transfert de fichiers afin de s'assurer que l'application reçoit tous les messages de progression et de journal.

Chacune de ces approches est décrite plus en détail dans les sections qui suivent.

Ne vous fiez pas au plug-in IBM MQ Explorer pour conserver les informations de journal.

Utilisation du consignateur de base de données Managed File Transfer pour conserver les messages de journal

Le consignateur de base de données est un composant facultatif de Managed File Transfer que vous pouvez utiliser pour copier des informations de journal dans une base de données à des fins d'analyse et d'audit. Le consignateur de base de données est une application Java autonome que vous installez sur un système qui héberge le gestionnaire de file d'attente de coordination et la base de données. Pour plus d'informations sur le consignateur de base de données, voir [«Configuration d'un consignateur MFT»](#), à la page 852.

Conservation des messages de progression et de journal à l'aide du plug-in IBM MQ Explorer

Lorsqu'une instance du plug-in IBM MQ Explorer est démarrée pour la première fois, elle crée un abonnement durable sur le gestionnaire de files d'attente de coordination. Cet abonnement durable est utilisé pour collecter les informations affichées dans les vues **Journal de transfert** et **Progression du transfert en cours** .

Le nom de l'abonnement durable est préfixé pour indiquer que l'abonnement a été créé par le plug-in IBM MQ Explorer MFT , le nom d'hôte et le nom de l'utilisateur. Par exemple, MQExplorer_MFT_Plugin_HOST_TJWatson.

Ce préfixe est ajouté si un administrateur souhaite supprimer un abonnement durable qui n'est plus utilisé par une instance du plug-in IBM MQ Explorer .

L'utilisation d'un abonnement durable sur le gestionnaire de files d'attente de coordination peut entraîner la génération de messages sur SYSTEM.MANAGED.DURABLE . Si vous disposez d'un réseau Managed File Transfer à volume élevé, utilisez rarement le plug-in IBM MQ Explorer , ou les deux, ces données de message peuvent remplir le système de fichiers local.

Pour éviter que cela ne se produise, indiquez que le plug-in IBM MQ Explorer utilise un abonnement non durable au gestionnaire de files d'attente de coordination. Effectuez les étapes suivantes dans votre instance IBM MQ Explorer :

1. Sélectionnez **la fenêtre** > **Préférences** > **Explorateur MQ** > **Transfert de fichier géré**
2. Dans la liste **Type d'abonnement au journal de transfert**, choisissez NON_DURABLE.

Stockage des publications dans une file d'attente IBM MQ

Pour stocker les messages de journal ou de progression dans une file d'attente IBM MQ , configurez un abonnement sur le gestionnaire de file d'attente de coordination qui achemine les messages vers cette file d'attente. Par exemple, pour réacheminer tous les messages de journal vers une file d'attente nommée LOG.QUEUE, soumettez la commande MQSC suivante:

```
define sub(MY.SUB) TOPICSTR('Log/#') TOPICOBJ(SYSTEM.FTE) DEST(LOG.QUEUE)WSHEMA(TOPIC)
```


Une fois que les messages de journal ont été réacheminés vers une file d'attente IBM MQ, ils sont conservés dans la file d'attente jusqu'à ce qu'ils soient traités par une application IBM MQ qui utilise la file d'attente.

Écriture d'applications qui gèrent un abonnement durable à SYSTEM.FTE

Vous pouvez écrire des applications qui gèrent leurs propres abonnements durables à SYSTEM.FTE à l'aide de l'une des interfaces de programme d'application prises en charge par IBM MQ. Ces applications peuvent recevoir des messages de file d'attente ou de journal IBM MQ et les traiter de manière appropriée en fonction de vos besoins métier.

Pour plus d'informations sur les interfaces de programme d'application disponibles, voir [Développement d'applications](#).

Configuration des gestionnaires de files d'attente d'agent MFT

Après l'installation, exécutez le script `agent_name_create.mqsc` dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/agents/agent_name` pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente d'agent. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez ces étapes sur le gestionnaire de files d'attente d'agent.

Procédure

1. Créez les files d'attente d'opérations de l'agent.

Ces files d'attente sont nommées:

- SYSTEME SYSTEM.FTE.COMMAND.*nom_agent*
- SYSTEME SYSTEM.FTE.DATA.*nom_agent*
- SYSTEME SYSTEM.FTE.EVENT.*nom_agent*
- SYSTEME SYSTEM.FTE.REPLY.*nom_agent*
- SYSTEME SYSTEM.FTE.STATE.*nom_agent*

Pour plus d'informations sur les paramètres de file d'attente et sur la façon dont les files d'attente sont utilisées, voir [MFT Paramètres de file d'attente d'agent](#).

2. Créez les files d'attente de droits d'accès de l'agent.

Ces files d'attente sont nommées:

- SYSTEM.FTE.AUTHADM1.*nom_agent*
- SYSTEM.FTE.AUTHAGT1.*nom_agent*
- SYSTEM.FTE.AUTHMON1.*nom_agent*
- SYSTEM.FTE.AUTHOPS1.*nom_agent*
- SYSTEM.FTE.AUTHSCH1.*nom_agent*
- SYSTEM.FTE.AUTHTRN1.*nom_agent*

Pour plus d'informations sur les paramètres de file d'attente et sur la façon dont les files d'attente sont utilisées, voir [MFT Paramètres de file d'attente d'agent](#).

Que faire ensuite

Pour plus d'informations sur la création et la configuration d'un agent de pont de protocole, voir [fteCreateBridgeAgent \(création et configuration d'un agent de pont de protocole MFT\)](#) et [Configuration d'un pont de protocole pour un serveur FTPS](#).

Concepts associés

«connexion à IBM MQ», à la page 831

Toutes les communications réseau avec les gestionnaires de files d'attente IBM MQ, y compris les communications liées à Managed File Transfer, impliquent des canaux IBM MQ. Un canal IBM MQ

représente une extrémité d'une liaison réseau. Les canaux sont classés en tant que canaux de message ou canaux MQI.

[«Configuration d'un gestionnaire de files d'attente multi-instance pour une utilisation avec MFT»](#), à la page 837

IBM WebSphere MQ 7.0.1 et versions ultérieures prennent en charge la création de gestionnaires de files d'attente multi-instance. Un gestionnaire de files d'attente multi-instance redémarre automatiquement sur un serveur de secours. Managed File Transfer prend en charge la connexion à des gestionnaires de files d'attente d'agent multi-instance, à un gestionnaire de files d'attente de coordination multi-instance et à un gestionnaire de files d'attente de commandes multi-instance.

Tâches associées

[«Configuration des gestionnaires de files d'attente réseau MFT»](#), à la page 832

Si votre réseau Managed File Transfer inclut plusieurs gestionnaires de files d'attente IBM MQ, ces gestionnaires de files d'attente IBM MQ doivent pouvoir communiquer à distance entre eux.

[«Configuration du gestionnaire de files d'attente de coordination pour MFT»](#), à la page 833

Après avoir exécuté la commande **fteSetupCoordination**, exécutez le script *coordination_qmgr_name.mqsc* dans le répertoire *MQ_DATA_PATH/mqft/config/coordination_qmgr_name* pour effectuer la configuration nécessaire pour le gestionnaire de files d'attente de coordination. Toutefois, si vous souhaitez effectuer cette configuration manuellement, effectuez les étapes suivantes sur le gestionnaire de files d'attente de coordination.

Référence associée

[Paramètres de file d'attente de l'agent MFT](#)

[fteSetupCoordination](#)

Configuration d'un agent MFT pour plusieurs canaux dans un cluster

Si vous souhaitez utiliser la prise en charge multicanal de IBM MQ dans une configuration en cluster, définissez d'abord la propriété **agentMultipleChannelsEnabled** sur `true`, puis effectuez les étapes de cette rubrique.

Pourquoi et quand exécuter cette tâche

Dans un cluster, la prise en charge multicanal est activée par les définitions IBM MQ sur le gestionnaire de files d'attente de l'agent de destination uniquement.

Vous devez effectuer les étapes de cette rubrique en plus des étapes de configuration IBM MQ standard requises pour un agent Managed File Transfer, qui sont répertoriées dans [«Configuration de MFT pour la première utilisation»](#), à la page 830.

Les exemples de configuration suivants utilisent les commandes **runmqsc**.

Procédure

1. Définissez un canal récepteur de cluster pour chaque canal que vous souhaitez utiliser. Par exemple, si vous utilisez deux canaux:

```
DEFINE CHANNEL(TO.DESTQMGRNAME_1) CHLTYPE(CLUSRCVR) CLUSTER(MFTCLUSTER)
DEFINE CHANNEL(TO.DESTQMGRNAME_2) CHLTYPE(CLUSRCVR) CLUSTER(MFTCLUSTER)
```

où :

- *DESTQMGRNAME* est le nom du gestionnaire de files d'attente de l'agent de destination.
- *MFTCLUSTER* est le nom du cluster IBM MQ.

Il est recommandé d'utiliser la convention de dénomination *MFTCLUSTER.DESTMGRNAME_n* pour les canaux, mais cette convention n'est pas obligatoire.

2. Définissez un alias de gestionnaire de files d'attente correspondant à chaque canal. Exemple :

```
DEFINE QREMOTE(SYSTEM.FTE.DESTQMGRNAME_1) RQMNAME(DESTQMGRNAME) CLUSTER(MFTCLUSTER)
DEFINE QREMOTE(SYSTEM.FTE.DESTQMGRNAME_2) RQMNAME(DESTQMGRNAME) CLUSTER(MFTCLUSTER)
```

Vous devez utiliser `SYSTEM.FTE.DESTQMGRNAME_n` convention de dénomination des alias de gestionnaire de files d'attente car l'agent émetteur recherche des alias de gestionnaire de files d'attente de ce format. Les nombres que vous utilisez pour `n` doivent commencer à 1 et être consécutifs. Vous devez définir les définitions à l'échelle du cluster pour qu'elles soient disponibles sur le gestionnaire de files d'attente de l'agent source.

Pour que l'agent source et l'agent de destination puissent déterminer correctement le nombre d'alias de gestionnaire de files d'attente, ne définissez **pas** de file d'attente XMITQ par défaut pour le gestionnaire de files d'attente.

Tâches associées

«Configuration d'un agent MFT pour plusieurs canaux: non configurés en cluster», à la page 843

Si vous souhaitez utiliser la prise en charge multicanal IBM MQ dans une configuration sans cluster, définissez d'abord la propriété `agentMultipleChannelsEnabled` sur `true`, puis effectuez les étapes de cette rubrique.

Référence associée

Le fichier `MFT agent.properties`

Configuration d'un agent MFT pour plusieurs canaux: non configurés en cluster

Si vous souhaitez utiliser la prise en charge multicanal IBM MQ dans une configuration sans cluster, définissez d'abord la propriété `agentMultipleChannelsEnabled` sur `true`, puis effectuez les étapes de cette rubrique.

Pourquoi et quand exécuter cette tâche

Dans une configuration sans cluster, la prise en charge multicanal est activée par les définitions IBM MQ sur le gestionnaire de files d'attente de l'agent source et de l'agent de destination.

Vous devez effectuer les étapes de cette rubrique en plus des étapes de configuration IBM MQ standard requises pour un agent Managed File Transfer, qui sont répertoriées dans «Configuration de MFT pour la première utilisation», à la page 830.

Les étapes suivantes supposent que les canaux émetteurs-récepteurs sont utilisés pour communiquer entre les gestionnaires de files d'attente source et de destination.

Les exemples de configuration suivants utilisent les commandes `runmqsc`.

Procédure

1. Sur le gestionnaire de files d'attente de l'agent de destination, définissez un canal récepteur pour chaque canal que vous souhaitez utiliser. Par exemple, si vous utilisez deux canaux:

```
DEFINE CHANNEL(TO.DESTQMGRNAME_1) CHLTYPE(RCVR) TRPTYPE(TCP)
DEFINE CHANNEL(TO.DESTQMGRNAME_2) CHLTYPE(RCVR) TRPTYPE(TCP)
```

où: `DESTQMGRNAME` est le nom du gestionnaire de files d'attente de l'agent de destination.

Il est recommandé d'utiliser `TO.DESTMGRNAME_n` pour les canaux, mais cette convention n'est pas obligatoire. Les noms de canal récepteur doivent correspondre aux canaux émetteurs correspondants sur le gestionnaire de files d'attente de l'agent source.

2. Sur le gestionnaire de files d'attente de l'agent source, définissez une file d'attente de transmission pour chaque canal que vous souhaitez utiliser. Par exemple, si vous utilisez deux canaux:

```
DEFINE QLOCAL (DESTQMGRNAME_1) USAGE(XMITQ)
DEFINE QLOCAL (DESTQMGRNAME_2) USAGE(XMITQ)
```

Il est recommandé d'utiliser la convention de dénomination DESTMGRNAME_n pour les files d'attente de transmission, mais cette convention n'est pas obligatoire. Les files d'attente de transmission que vous définissez sont référencées à partir des définitions de canal émetteur et des définitions d'alias de gestionnaire de files d'attente dans les étapes suivantes.

3. Sur le gestionnaire de files d'attente de l'agent source, définissez un canal émetteur pour chaque canal que vous souhaitez utiliser. Par exemple, si vous utilisez deux canaux:

```
DEFINE CHANNEL (TO.DESTQMGRNAME_1) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(DESTHOST:port)
XMITQ(DESTQMGRNAME_1)
DEFINE CHANNEL (TO.DESTQMGRNAME_2) CHLTYPE(SDR) TRPTYPE(TCP) CONNAME(DESTHOST:port)
XMITQ(DESTQMGRNAME_2)
```

Il est recommandé d'utiliser TO.DESTMGRNAME_n pour les canaux, mais cette convention n'est pas obligatoire. Les noms de canal émetteur doivent correspondre aux canaux récepteurs correspondants sur le gestionnaire de files d'attente de l'agent de destination.

4. Sur le gestionnaire de files d'attente de l'agent source, définissez un alias de gestionnaire de files d'attente correspondant à chaque canal. Exemple :

```
DEFINE QREMOTE (SYSTEM.FTE.DESTQMGRNAME_1) RQMNAME (DESTQMGRNAME) XMITQ (DESTQMGRNAME_1)
DEFINE QREMOTE (SYSTEM.FTE.DESTQMGRNAME_2) RQMNAME (DESTQMGRNAME) XMITQ (DESTQMGRNAME_2)
```

Vous devez utiliser SYSTEM.FTE.DESTQMGRNAME_n pour les alias de gestionnaire de files d'attente car l'agent émetteur recherche les alias de gestionnaire de files d'attente de ce format. Les nombres que vous utilisez pour n doivent commencer à 1 et être consécutifs.

Pour que l'agent puisse déterminer correctement le nombre d'alias de gestionnaire de files d'attente, ne définissez **pas** de file d'attente XMITQ par défaut pour le gestionnaire de files d'attente.

Tâches associées

«Configuration d'un agent MFT pour plusieurs canaux dans un cluster», à la page 842

Si vous souhaitez utiliser la prise en charge multicanal de IBM MQ dans une configuration en cluster, définissez d'abord la propriété **agentMultipleChannelsEnabled** sur true, puis effectuez les étapes de cette rubrique.

Référence associée

Le fichier MFT [agent.properties](#)

Configuration des agents MFT avec MSCS

La configuration de l'agent Managed File Transfer (MFT) Microsoft Cluster Service (MSCS) est prise en charge si la plateforme est prise en charge par MFT et qu'elle exécute l'une des versions de Windows.

Pourquoi et quand exécuter cette tâche

Cette tâche décrit deux scénarios que vous pouvez suivre pour effectuer la reprise en ligne d'un agent MFT :

- Scénario 1: Configuration de l'agent en tant que ressource MSCS.
- Scénario 2: Configuration du gestionnaire de files d'attente d'agent et de l'agent en tant que ressources MSCS.

Procédure

Scénario 1: Configuration de l'agent en tant que ressource MSCS

- Pour configurer l'agent en tant que ressource MSCS, procédez comme suit:
 - a) Installez Managed File Transfer localement sur chaque machine du cluster.

Voir Installation de Managed File Transfer.

- b) Créez l'agent sur la machine principale du cluster.

L'agent doit être configuré pour se connecter au gestionnaire de files d'attente d'agent à l'aide du transport CLIENT. Veillez à créer tous les objets sur le gestionnaire de files d'attente pour cet agent. Pour plus d'informations sur la procédure à suivre, voir [Configuration de l'agent](#).

- c) Modifiez l'agent pour qu'il s'exécute en tant que service Windows et configurez-le de sorte qu'il ne démarre pas automatiquement lorsque Windows redémarre en définissant la zone **Type de démarrage** du service d'agent dans l'outil Services Windows sur Manuel.

Pour plus d'informations, voir [Démarrage d'un agent MFT en tant que service Windows](#).

- d) Répétez l'étape «2», à la page 845 et l'étape «3», à la page 845 du scénario 1 sur la machine secondaire.

Cela garantit que la structure de fichiers pour les journaux, les propriétés, etc., existe sur l'autre machine du cluster. Notez qu'il n'est pas nécessaire de créer les objets de gestionnaire de files d'attente comme à l'étape «2», à la page 845.

- e) Sur la machine principale, ajoutez l'agent en tant que 'Service générique' sous contrôle MSCS.

Pour cela, procédez comme suit :

- a. Cliquez avec le bouton droit de la souris sur le cluster et sélectionnez **Rôle-> Ajouter une ressource->'Service générique'**.
- b. Dans la liste des services Windows , sélectionnez le service d'agent et exécutez l'assistant de configuration en cliquant sur **Suivant**.

Le service d'agent est désormais ajouté en tant que ressource MSCS. Si une reprise en ligne se produit, le service d'agent est démarré sur l'autre machine.

Scénario 2: Configuration du gestionnaire de files d'attente d'agent et de l'agent en tant que ressources MSCS

- Pour configurer le gestionnaire de files d'attente d'agent et l'agent en tant que ressources MSCS, procédez comme suit:

- a) Configurez le gestionnaire de files d'attente d'agent pour qu'il s'exécute en tant que ressource MSCS.

Pour plus d'informations sur la procédure à suivre, voir [«Placement d'un gestionnaire de files d'attente sous contrôle MSCS», à la page 535](#).

- b) Créez l'agent sur la machine principale du cluster.

L'agent doit être configuré pour se connecter au gestionnaire de files d'attente d'agent à l'aide du transport BINDINGS. Veillez à créer tous les objets sur le gestionnaire de files d'attente pour cet agent. Pour plus d'informations sur la procédure à suivre, voir [Configuration de l'agent](#).

- c) Modifiez l'agent pour qu'il s'exécute en tant que service Windows et configurez-le de sorte qu'il ne démarre pas automatiquement lorsque Windows redémarre en définissant la zone **Type de démarrage** du service d'agent dans l'outil Services Windows sur Manuel.

Pour plus d'informations, voir [Démarrage d'un agent MFT en tant que service Windows](#).

- d) Vérifiez que le gestionnaire de files d'attente d'agent (qui est sous le contrôle de MSCS) est en cours d'exécution sur la machine secondaire.

L'agent créé sur cette machine se connecte au gestionnaire de files d'attente à l'aide du transport BINDINGS et doit donc être disponible lors de la création de l'agent.

- e) Répétez l'étape «2», à la page 845 et l'étape «3», à la page 845 du scénario 2 sur la machine secondaire.

Cela garantit que la structure de fichiers pour les journaux, les propriétés, etc., existe sur l'autre machine du cluster. Notez qu'il n'est pas nécessaire de créer les objets de gestionnaire de files d'attente comme à l'étape «2», à la page 845.

- f) Ajoutez l'agent en tant que 'service générique' sous le contrôle de MSCS.

Pour cela, procédez comme suit :

- a. Cliquez avec le bouton droit de la souris sur le cluster et sélectionnez **Rôle-> Ajouter une ressource->'Service générique'**.
- b. Dans la liste des services Windows , sélectionnez le service d'agent et exécutez l'assistant de configuration en cliquant sur **Suivant**.
- g) Modifiez les propriétés de ressource du service d'agent pour ajouter la ressource de gestionnaire de files d'attente dans la liste des dépendances.
Cela garantit que la ressource du gestionnaire de files d'attente est démarrée avant le démarrage de l'agent.
- h) Mettez la ressource du gestionnaire de files d'attente hors ligne, puis mettez la ressource de l'agent en ligne. Vérifiez si la ressource du gestionnaire de files d'attente et l'agent sont démarrés.
Si une reprise en ligne se produit, le service d'agent et le gestionnaire de files d'attente d'agent sont démarrés sur la machine secondaire.

Agents à haute disponibilité dans Managed File Transfer

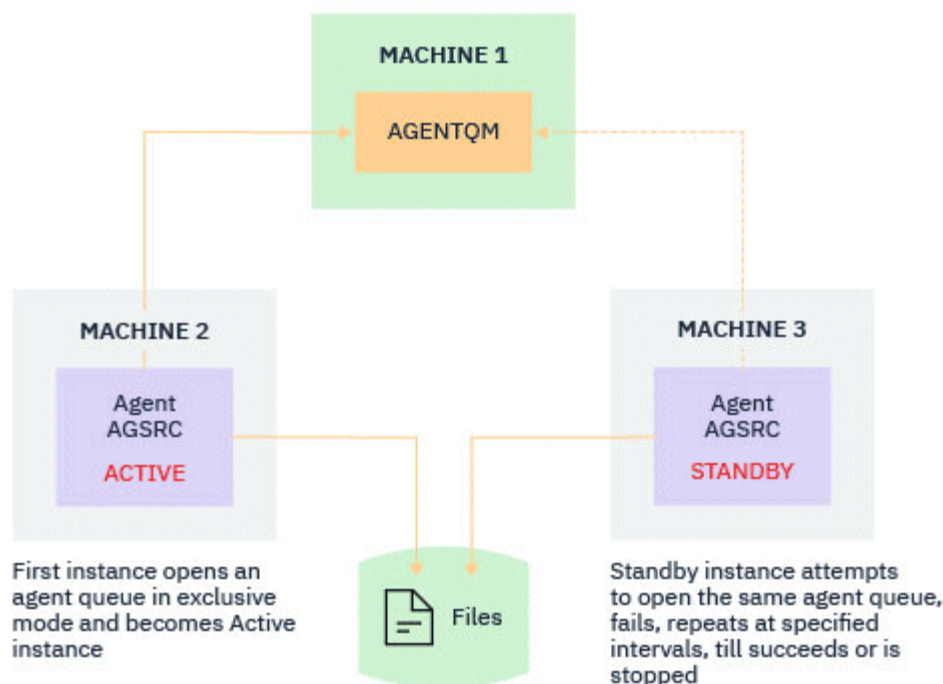
Vous pouvez configurer des agents standard ou de pont dans MFT pour qu'ils s'exécutent dans une configuration à haute disponibilité. Une paire d'instances d'agent avec des configurations identiques sont impliquées dans la configuration à haute disponibilité, où une instance s'exécute sur une machine alors qu'une autre instance s'exécute sur une autre machine. Les deux instances sont configurées pour se connecter au même gestionnaire de files d'attente d'agent.

Présentation

Une seule des deux instances, appelée *instance active*, traite les transferts de fichiers, tandis que l'autre instance, appelée *instance de secours*, est partiellement initialisée et ne peut pas traiter les transferts de fichiers.

Lorsqu'une instance active échoue ou perd la connectivité au gestionnaire de files d'attente, l'instance de secours termine son initialisation, devient active et démarre le traitement des transferts de fichiers. Tous les transferts en cours lors de l'échec de la reprise de l'instance active à partir du dernier point de contrôle connu.

L'illustration suivante montre une configuration commune des agents actifs et de



secours:

Remarques :

1. Une instance d'un agent s'exécute sur deux machines différentes, l'une étant une *instance active* et l'autre une *instance de secours*.
2. Chaque instance de l'agent s'exécute sur une machine différente, avec l'une des instances en tant qu'instance active et l'autre en tant qu'instance de secours.
3. Le même ensemble de files d'attente d'agent est partagé entre les deux instances de l'agent.
4. Les deux instances de l'agent doivent accéder au même système de fichiers partagé pour effectuer des transferts gérés.

Le mécanisme de l'instance d'agent active-de secours fonctionne en prenant un verrou sur une ressource partagée. L'instance d'agent qui prend un verrou sur la ressource partagée devient l'instance active tandis que l'autre instance (qui ne prend pas de verrou) devient une instance de secours.

La ressource partagée ici est une nouvelle file d'attente, SYSTEM.FTE.HA.<agent name>. Cette file d'attente est créée automatiquement lorsqu'un agent IBM MQ 9.1.4 ou version ultérieure est configuré.

Fonctionnement du processus

Pour créer un agent à haute disponibilité, vous créez un agent avec des paramètres de configuration identiques sur deux machines en exécutant la commande **fteCreateAgent** ou **fteCreateBridgeAgent** à l'aide du paramètre **-x** supplémentaire et de la propriété d'agent **highlyAvailable** dans le fichier agent.properties défini sur true.

Remarques :

- Les deux configurations doivent pointer vers le même gestionnaire de files d'attente d'agent.
- Les files d'attente d'agent requises ne doivent être créées qu'une seule fois sur le gestionnaire de files d'attente d'agent.

Voir la commande **fteCreateAgent** pour plus d'informations sur le paramètre **-x** et le fichier agent.properties pour plus d'informations sur la propriété d'agent **highlyAvailable**.

Remarque : L'exécution de la commande **fteCreateAgent** ou **fteCreateBridgeAgent** crée un fichier MQSC contenant les scripts requis pour créer des objets IBM MQ dans le gestionnaire de files d'attente de l'agent et la file d'attente SYSTEM.FTE.HA.agent name. Ce fichier MQSC est créé, que vous indiquiez ou non le paramètre **-x**.

Lors de la création d'une configuration d'agent à haute disponibilité, la commande **fteCreateAgent** ou **fteCreateBridgeAgent** vérifie l'existence d'une instance du même agent présente ailleurs en s'abonnant à la rubrique SYSTEM.FTE/Agents/agent name. Si une instance du même agent est trouvée, l'une des commandes crée la configuration requise sur le système de fichiers mais ne publie pas à nouveau la création de l'agent.

Lorsqu'un agent démarre en mode haute disponibilité:

1. L'agent tente d'ouvrir la file d'attente SYSTEM.FTE.HA.agent name en mode GET exclusif.
2. Si l'agent ouvre correctement la file d'attente SYSTEM.FTE.HA.agent name, elle devient l'instance active d'un agent et un nouveau processus de démarrage se poursuit.
3. Si la tentative d'ouverture de la file d'attente SYSTEM.FTE.HA.agent name en mode GET exclusif échoue avec le code anomalie MQRC_OBJECT_IN_USE, cela signifie qu'une instance active de l'agent est déjà en cours d'exécution ailleurs. Par conséquent, cette instance devient l'instance de secours de l'agent.

L'instance de secours tente d'ouvrir la file d'attente SYSTEM.FTE.HA.agent name à des intervalles spécifiés. Une propriété d'agent supplémentaire **standbyPollInterval** est fournie à cet effet dans le fichier agent.properties.

Avec la valeur par défaut, l'instance de secours tente d'ouvrir la file d'attente SYSTEM.FTE.HA.agent name toutes les cinq secondes. Cette opération se répète jusqu'à ce que l'instance parvienne à ouvrir la file d'attente SYSTEM.FTE.HA.agent name ou qu'elle soit arrêtée à l'aide de la commande **fteStopAgent**.

La propriété **standbyPollInterval** est également utilisée par toutes les instances pour déterminer combien de temps une instance attend entre deux tentatives de reconnexion si elle se déconnecte de son gestionnaire de files d'attente d'agent.

Plusieurs instances de secours

Toutes les instances de secours tentent de prendre la file d'attente `SYSTEM.FTE.HA.agent name` en mode GET exclusif et l'instance qui aboutit, après l'échec de l'instance active, devient l'instance active.

L'instance active gère les informations de toutes les instances de secours connues et publie les informations dans le cadre de la publication du statut de l'agent. La sortie de la commande **fteShowAgentDetails**, la réponse de l'agent GET REST API et le plug-in IBM MQ Explorer MFT affichent des informations sur toutes les instances de secours.

Pour plus d'informations, consultez les exemples de sortie de la commande **fteShowAgentDetails** et de la réponse GET REST API de l'agent.

Voir Messages d'état de l'agentMFT pour des exemples d'informations d'état de l'agent au format XML.

Version requise

Les agents actifs et de secours doivent être IBM MQ 9.1.4 ou supérieurs.



Avertissement :

- Vous ne pouvez pas configurer ou démarrer des versions d' IBM MQ antérieures à IBM MQ 9.1.4 en mode haute disponibilité.
- Les instances actives et de secours doivent exécuter la même version du code.

La version des instances active et de secours est validée pour s'assurer que les deux instances sont de la même version. Une file d'attente dynamique temporaire est utilisée pour la communication entre les instances. Deux propriétés d'agent, **dynamicQueuePrefix** et **modelQueueName**, définies dans le fichier `agent.properties`, génèrent le nom de la file d'attente dynamique temporaire.

Informations requises pour les agents à haute disponibilité dans Managed File Transfer

Il existe différents types d'informations que vous devez connaître sur les agents MFT standard ou de pont qui s'exécutent dans une configuration à haute disponibilité. Ces informations incluent les différentes méthodes de démarrage de l'agent, la manière d'identifier l'instance de l'agent dans le fichier journal et les informations de statut de l'agent.

Démarrage d'un agent

Une instance d'un agent s'exécute ailleurs en mode sans haute disponibilité

Si vous tentez de démarrer une autre instance de l'agent qui n'est pas configurée en tant qu'agent à haute disponibilité, vous devez d'abord vérifier si un verrou peut être acquis sur la file d'attente `SYSTEM.FTE.HA.agent name`.

Comme l'autre instance a été démarrée en mode sans haute disponibilité, le verrou de la file d'attente `SYSTEM.FTE.HA.agent name` sera acquis par cette instance. L'agent poursuit l'initialisation, mais échoue ultérieurement car la file d'attente de commandes est ouverte exclusivement par une autre instance.

Dans ce cas, les messages affichés dans l'exemple suivant sont consignés dans le fichier `output0.log` de l'agent et celui-ci poursuit sa tentative d'ouverture de la file d'attente de commandes toutes les 30 secondes:

```
BFGMQ1045I: File d'attente système de l'agent'SYSTEM.FTE.COMMAND.SRC'est configuré en tant que  
NOSHARE ou  
DEFSOPT (SHARED).
```

```
BFGAG0035W: L'agent a reçu le code anomalie MQI 2042 lors de la tentative d'ouverture de la file  
d'attente
```


'SYSTEM.FTE.COMMAND.SRC' sur le gestionnaire de files d'attente 'MFTHAQM' avec le nom de connexion 'localhost (1414)' et le canal 'MFT_HA_CHN'. L'agent essaiera à nouveau l'opération toutes les 30 secondes.

Une instance d'un agent s'exécute en mode haute disponibilité ailleurs

Si vous tentez de démarrer une autre instance de l'agent qui n'est pas configurée en tant qu'agent à haute disponibilité, vous devez d'abord vérifier si un verrou peut être acquis sur la file d'attente SYSTEM.FTE.HA.*agent name*.

Etant donné que l'autre instance s'exécute en tant qu'instance active, la tentative d'acquisition d'un verrou échoue. Le démarrage de l'instance échoue et le message d'erreur suivant est consigné dans le fichier output0.log de l'agent:

```
BFGAG0194E: Une instance de cet agent est déjà en cours d'exécution ailleurs.  
Par conséquent, cette instance ne peut pas continuer et  
va être arrêtée.
```

Démarrage de l'agent en tant que service Windows

Sous Windows, vous pouvez démarrer un agent en tant que service Windows.

Lors du démarrage, Windows démarre l'agent MFT en mode normal ou haute disponibilité. Si l'agent est configuré pour s'exécuter en mode haute disponibilité, le service s'exécute en tant qu'instance active ou de secours, en fonction de l'instance qui acquiert le verrou en premier.

Identification du type d'instance d'un agent dans le fichier journal

Les messages d'information sont écrits dans le fichier output0.log de l'agent pour indiquer le type d'instance. Lorsqu'une instance d'agent démarre en tant qu'instance active, le message suivant est consigné:

```
BFGAG0193I: L'agent a été initialisé en tant qu'instance active.
```

Lorsqu'une instance d'agent démarre en tant qu'instance de secours, le message suivant s'affiche:

```
BFGAG0193I: L'initialisation de l'agent en tant qu'instance de secours a abouti.
```

Mises à jour du statut des agents

Comme deux instances du même agent sont en cours d'exécution, vous devez disposer des informations relatives aux deux instances dans la publication du statut de l'agent.

Notez que l'instance active est celle qui publie le statut des deux instances.

Instance de secours

Lors de la publication du statut de l'agent, l'instance active vérifie l'ancienneté de la publication de l'instance de secours.

Le fichier [agent.properties](#) contient deux propriétés supplémentaires à cet effet:

- **standbyStatusExpiry** est le délai d'expiration du message d'état de secours à insérer dans la file d'attente de commandes de l'agent. Le message expire si l'instance active d'un agent ne traite pas ce message pendant cette période.

Par défaut, la valeur de **standbyStatusExpiry** est de 30 secondes. Le message est également un message de faible priorité, 9, pour permettre le traitement prioritaire des demandes de transfert par rapport aux messages d'état de secours.

- **standbyStatusPublishInterval** définit la fréquence à laquelle l'instance de secours publie son état.

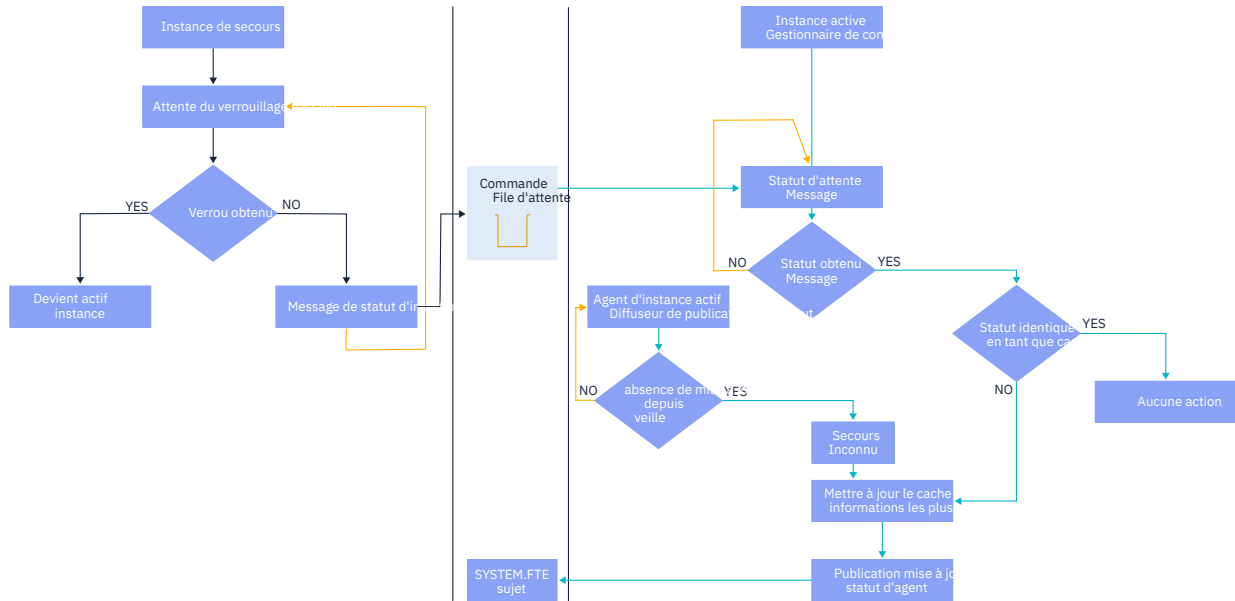
Instance active

L'instance active effectue les opérations suivantes pour traiter les mises à jour de statut à partir de l'instance de secours:

1. Extrait le message de la file d'attente SYSTEM.FTE.COMMAND.<agent name> et délègue le traitement des messages à une unité d'exécution de tâche.
2. L'unité d'exécution de tâche extrait le contenu du corps du message, met à jour l'objet de statut de l'agent avec les informations d'instance de secours et indique au diffuseur de publications de statut de l'agent de publier le statut.
3. L'éditeur de statut de l'agent publie le statut.

Notez que les optimisations sont effectuées ici pour mettre en cache les informations de statut de secours. Lorsqu'une demande est effectuée, le diffuseur de publications de statut de l'agent vérifie le nouveau statut avec le statut mis en cache et publie uniquement s'il existe une différence.

Le diagramme suivant décrit le flux suivi par les instances actives ou de secours pour publier le statut d'un agent:



Suppression des instances, de la reprise en ligne et de la maintenance dans les agents à haute disponibilité

Les instances Managed File Transfer à haute disponibilité peuvent être supprimées, échouer de différentes manières et nécessiter une maintenance.

Suppression du statut de l'instance de secours

Dans certains cas, l'instance active est occupée par des transferts et ne peut pas traiter les messages d'état de l'instance de secours, ou l'instance de secours a échoué ou ne publie pas de messages d'état pour une raison quelconque.

Dans de tels scénarios, l'agent actif qui était conscient de la présence d'une instance de secours attend la valeur spécifiée par la propriété **standbyStatusDiscardTime** dans le fichier `agent.properties` avant de supprimer l'instance de secours de sa liste. La valeur par défaut de cette propriété est de 600 secondes, soit le double de celle de la propriété **standbyStatusPublishInterval**.

Reprise d'une instance en cas de défaillance normale

Vous devez utiliser la commande **fteStopAgent** avec l'option **-i** pour effectuer une reprise en ligne normale.

Cela permet de s'assurer que l'instance active est arrêtée immédiatement. Si vous arrêtez un agent sans l'option **-i**, l'agent continue de s'exécuter jusqu'à ce que tous les transferts en cours soient terminés par l'instance active. Par conséquent, la reprise en ligne peut prendre un certain temps.

Tout transfert en cours reprend à partir du dernier point de contrôle connu.

Reprise d'une instance dans d'autres situations

Si une instance active se termine d'une manière qui n'est pas normale, ou si l'ensemble de la machine échoue, la connexion à la file d'attente de l'agent est interrompue et le gestionnaire de files d'attente ferme toutes les files d'attente ouvertes, y compris la file d'attente SYSTEM.FTE.HA.<agent name>, et les connexions.

Pour cette raison, l'instance de secours acquiert l'opération GET exclusive et termine le reste de l'initialisation de l'agent.

Là encore, tous les transferts en cours reprennent à partir des derniers points de contrôle connus.

Si une connexion au gestionnaire de files d'attente est interrompue

Mode client

Un processus agent se compose de plusieurs unités d'exécution. Outre les unités d'exécution par défaut, par exemple, une unité d'exécution qui publie le statut de l'agent à intervalles réguliers, chaque demande de transfert est traitée avec un ensemble d'unités d'exécution qui se terminent à la fin d'un transfert.

La plupart de ces unités d'exécution se connectent au gestionnaire de files d'attente de l'agent et placent et extraient des messages. Il est possible que ces connexions soient interrompues en raison d'un problème réseau ou d'un échec du gestionnaire de files d'attente. Lorsqu'une unité d'exécution détecte un problème de connexion interrompue, elle informe l'unité d'exécution principale qu'elle doit lancer la reprise et s'arrête.

L'unité d'exécution principale lance ensuite une autre unité d'exécution pour attendre l'établissement d'une connexion au gestionnaire de files d'attente. Une fois reconnecté, une tentative d'acquisition de l'opération GET exclusive pour l'agent est effectuée. Si cette opération aboutit, l'agent poursuit la récupération et devient l'instance active. Si la tentative d'acquisition de l'opération GET exclusive échoue, l'instance devient une instance de secours.

Mode liaison

Lors de la connexion en mode liaisons, si un agent perd la connexion, le processus de l'agent se termine. Le contrôleur de processus gère le redémarrage de l'agent. Lorsqu'un agent redémarre, il passe par le processus de tentative d'acquisition de l'opération GET exclusive pour lui-même.

Si l'agent réussit, il devient une instance active ; sinon, l'agent devient une instance de secours.

Application des mises à niveau du niveau de maintenance

Les étapes d'application de la maintenance aux agents à haute disponibilité sont similaires à celles décrites pour les gestionnaires de files d'attente multi-instance. Pour plus d'informations, voir [Application de mises à jour du niveau de maintenance à des gestionnaires de files d'attente multi-instance sur Windows](#) ou [Application de mises à jour du niveau de maintenance à des gestionnaires de files d'attente multi-instance sur AIX](#) ou [Application de mises à jour du niveau de maintenance à des gestionnaires de files d'attente multi-instance sur Linux](#).

Vous devez arrêter l'agent en cours d'exécution sur la machine sur laquelle le niveau de maintenance doit être appliqué, avant d'appliquer la maintenance. Si vous mettez à jour une instance active, pour assurer la continuité des transferts, vous devez basculer l'instance active vers une instance de secours.

Une fois la mise à niveau terminée, vous devez démarrer l'instance d'agent, basculer l'instance active en cours sur l'instance mise à niveau, puis mettre à niveau l'instance de secours.


Migration d'agents à partir d'une version antérieure du produit

Les agents migrés depuis des versions d'IBM MQ antérieures à IBM MQ 9.1.4 ne sont pas hautement disponibles. Vous pouvez les exécuter en mode haute disponibilité en suivant la procédure décrite dans [Migration des agents Managed File Transfer à partir d'une version antérieure](#).

Configuration d'un consignateur MFT

Lorsque Managed File Transfer transfère des fichiers, il publie des informations sur ses actions dans une rubrique du gestionnaire de files d'attente de coordination. Le consignateur de base de données est un composant facultatif de Managed File Transfer que vous pouvez utiliser pour copier ces informations dans une base de données à des fins d'analyse et d'audit.

Il existe trois versions du consignateur:

-  ALW consignateur de fichiers autonome
- consignateur de base de données autonome
- Consignateur Java Platform, Enterprise Edition (Java EE)

Consignateurs sous IBM i




Les programmes de consignment Managed File Transfer ne sont pas pris en charge sur la plateforme IBM i.

Consignateur de fichiers autonome



Le consignateur de fichiers autonome est un processus Java qui s'exécute sur le système qui héberge le gestionnaire de files d'attente de coordination ou sur un système qui héberge un gestionnaire de files d'attente avec une connectivité au gestionnaire de files d'attente de coordination. Le consignateur de fichiers autonome utilise des liaisons IBM MQ pour se connecter à son gestionnaire de files d'attente associé. Le consignateur autonome est créé à l'aide de la commande **fteCreateLogger**.

 **Windows** Vous pouvez exécuter le consignateur de fichiers autonome en tant que service Windows pour vous assurer que le consignateur de fichiers continue de s'exécuter lorsque vous vous déconnectez de votre session Windows et qu'il peut être configuré pour démarrer automatiquement lorsqu'un système redémarre. Pour plus d'informations, voir [«Installation du consignateur de fichiers MFT autonome»](#), à la page 853.

Le consignateur de fichiers autonome n'est pas pris en charge sur les plateformes suivantes:

-  z/OS
-  IBM i

Consignateur de base de données autonome

Le consignateur de base de données autonome est une application Java que vous installez sur un système qui héberge un gestionnaire de files d'attente et une base de données. Le consignateur de base de données autonome est souvent installé sur le même système que le gestionnaire de files d'attente de coordination, mais il peut également être installé sur le même système que tout gestionnaire de files d'attente disposant d'une connectivité au gestionnaire de files d'attente de coordination. Le consignateur de base de données autonome utilise des liaisons IBM MQ pour se connecter à son gestionnaire de files d'attente associé et un pilote JDBC de type 2 ou 4 pour se connecter à une base de données Db2 ou Oracle. Ces types de connexion sont requis car le consignateur de base de données autonome utilise la prise en charge XA du gestionnaire de files d'attente pour coordonner une transaction globale sur le gestionnaire de files d'attente et la base de données, protégeant ainsi les données.

Windows

Si vous utilisez un système Windows , vous pouvez exécuter les consigneurs autonomes en tant que services Windows pour vous assurer que les consigneurs continuent de s'exécuter lorsque vous vous déconnectez de votre session Windows . Pour plus d'informations, voir [«Installation du consigneur de base de données autonome MFT»](#), à la page 861 pour un consigneur de base de données autonome.

Java EE CONSIGNATEUR DE BASE DE DONNEES

Le consigneur de base de données Java EE est fourni sous la forme d'un fichier EAR que vous installez sur un serveur d'applications. Cela peut être plus pratique que d'utiliser le consigneur de base de données autonome si vous disposez d'un environnement de serveur d'applications Java EE existant car le consigneur de base de données Java EE peut être géré en même temps que vos autres applications d'entreprise. Vous pouvez également installer le consigneur de base de données Java EE sur un système distinct sur les systèmes hébergeant votre serveur et votre base de données IBM MQ . Le consigneur de base de données Java EE est pris en charge pour une utilisation avec les bases de données Db2 et Oracle . Le consigneur de base de données Java EE prend également en charge Oracle Real Application Clusters lorsqu'il est installé sous WebSphere Application Server 7.0.

Pour des instructions sur la configuration d'un consigneur, voir les rubriques suivantes:

- [«Installation du consigneur de fichiers MFT autonome»](#), à la page 853
- [«Installation du consigneur de base de données autonome MFT»](#), à la page 861
- [«Installation du consigneur de base de données Java EE pour MFT»](#), à la page 866

Tâches associées

[«Utilisation de MFT avec une base de données distante»](#), à la page 863

Vous pouvez utiliser le consigneur Managed File Transfer pour communiquer avec une base de données sur un système distant.

Référence associée

[Traitement des erreurs du consigneur MFT et rejet des messages](#)

[Propriétés de configuration du consigneur MFT](#)

ALW

Installation du consigneur de fichiers MFT autonome

Le consigneur de fichiers autonome est un processus Java qui peut se connecter à un gestionnaire de files d'attente de coordination à l'aide du mode de liaison IBM MQ ou du mode client. Pour définir un consigneur de fichiers autonome, utilisez la commande **fteCreateLogger** et suivez les étapes de cette rubrique.

Pourquoi et quand exécuter cette tâche


Pour plus d'informations sur le consigneur de fichiers autonome, voir [«Configuration d'un consigneur MFT»](#), à la page 852. Les étapes de cette rubrique permettent de configurer un consigneur pour qu'il se connecte à un gestionnaire de files d'attente de coordination. Pour d'autres configurations de consigneur, voir [«Autres configurations pour un consigneur MFT autonome»](#), à la page 865.

Le consigneur de fichiers autonome n'est pas pris en charge sur les plateformes suivantes:

-  z/OS
-  IBM i

Procédure

1. Vérifiez que le composant Managed File Transfer Logger est installé. Pour plus d'informations, voir [Options du produit Managed File Transfer](#).

2. Exécutez la commande **fteCreateLogger** en spécifiant le gestionnaire de files d'attente de coordination et en définissant le paramètre **-loggerType** sur FILE pour créer votre consignateur de fichiers autonome.
Pour plus d'informations, voir [fteCreateLogger](#).
3. Facultatif : Si vous souhaitez utiliser un format personnalisé, vous pouvez modifier le fichier XML créé par la commande **fteCreateLogger** . La définition du format de journal se trouve dans le fichier `FileLoggerFormat.xml` .
Pour plus d'informations, voir [«Format du consignateur de fichiers MFT autonome»](#), à la page 855.
4. Exécutez les commandes MQSC, fournies par la commande **fteCreateLogger** , sur votre gestionnaire de files d'attente de coordination pour créer les files d'attente du consignateur.
5. Identifiez un utilisateur pour exécuter le processus de consignateur et configurez les droits de cet utilisateur.
Pour plus d'informations, voir [«Configuration de l'accès utilisateur pour un consignateur de fichiers MFT autonome»](#), à la page 860.
6. Facultatif : Configurez davantage le consignateur de fichiers autonome en éditant le fichier `logger.properties` créé lors de l'exécution de la commande **fteCreateLogger** .
Ce fichier est un fichier de propriétés Java qui se compose de paires clé-valeur. Le fichier `logger.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name` . Pour plus d'informations sur les propriétés disponibles et leurs effets, voir [Propriétés de configuration du consignateurMFT](#).
7.  **Windows**
Facultatif : Si vous utilisez un système Windows , exécutez le consignateur de fichiers autonome en tant que service Windows .
Exécutez la commande **fteModifyLogger** avec le paramètre **-s** . Pour plus d'informations, voir [fteModifyLogger](#).
8. Démarrez le consignateur de fichiers autonome à l'aide de la commande **fteStartLogger** .
Pour plus d'informations, voir [fteStartLogger](#).
Si vous avez effectué l'étape précédente et utilisé la commande **fteModifyLogger** avec le paramètre **-s** sous Windows, le consignateur de fichiers autonome démarre en tant que service Windows .
9. Vérifiez la sortie du consignateur. Le consignateur de fichiers autonome génère deux types de sortie: les données d'audit de transfert de fichiers et les données de diagnostic du consignateur.
Les données d'audit de transfert de fichier se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name/logs`. Les données de diagnostic du consignateur se trouvent dans `MQ_DATA_PATH/mqft/logs/coordination_qmgr_name/loggers/logger_name`.
10. Arrêtez le consignateur à l'aide de la commande **fteStopLogger** .
Pour plus d'informations, voir [fteStopLogger](#).

Résultats

Tâches associées

«Configuration de l'accès utilisateur pour un consignateur de fichiers MFT autonome», à la page 860
Dans un environnement de test, vous pouvez ajouter tous les nouveaux privilèges nécessaires à votre compte utilisateur normal. Dans un environnement de production, il est recommandé de créer un utilisateur avec les droits minimum requis pour effectuer le travail.

Référence associée

[Propriétés de configuration du consignateur MFT](#)

[fteStartLogger \(démarrage d'un consignateur MFT\)](#)

[fteCreateLogger \(création d'un fichier MFT ou d'un consignateur de base de données\)](#)

[fteModifyLogger \(exécutez un consignateur MFT en tant que service Windows\)](#)

[fteStopLogger \(arrêt d'un consignateur MFT\)](#)

«Format du consignateur de fichiers MFT autonome», à la page 855

Le format des informations de message écrites par le consignateur de fichiers peut être défini dans le fichier `FileLoggerFormat.xml`.

Autorisations pour le consignateur MFT

ALW *Format du consignateur de fichiers MFT autonome*

Le format des informations de message écrites par le consignateur de fichiers peut être défini dans le fichier `FileLoggerFormat.xml`.

Le répertoire de configuration du consignateur se trouve dans `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`. Lors de la création d'un nouveau consignateur de fichiers, une version de ce fichier est créée et contient un ensemble par défaut de définitions utilisées par le consignateur de fichiers. Pour plus d'informations sur la définition du format de journal par défaut, voir MFT autonome file logger default log format.

Si vous souhaitez spécifier votre propre format de journal personnalisé, éditez le fichier `FileLoggerFormat.xml`.

Une définition de format de journal personnalisée

Une définition de format de journal se compose d'un ensemble de types de message, chaque type de message ayant une définition de format. Une définition de format pour un type de message est constituée d'un ensemble d'insertions fournies au format XPATH et d'un séparateur utilisé pour séparer chaque insertion. L'ordre des insertions détermine l'ordre dans lequel le contenu est placé dans les lignes générées pour la sortie dans les fichiers journaux. Par exemple, voici la définition du type de message `callStarted` :

```
<callStarted>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false">/transaction/action/
        @time</insert>
      <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
      <insert type="system" width="6" ignoreNull="false">type</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/agent/
        @agent</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/agent/@QMgr</insert>
      <insert type="user" width="0" ignoreNull="false">/transaction/job/name</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/
        call/command/@type</insert>
      <insert type="user" width="0" ignoreNull="true">/transaction/transferSet/
        call/command/@name</insert>
      <insert type="system" width="0" ignoreNull="true">callArguments</insert>
    </inserts>
    <separator></separator>
  </format>
</callStarted>
```

Ce format génère une ligne dans le fichier journal comme suit:

```
2011-11-25T10:53:04;414d5120514d5f67627468696e6b20206466c14e20004f02; [CSTR];
AGENT1;AGENT_QM;Managed Call;executable;echo;call test;
```

Les insertions fournies dans la définition de format sont dans l'ordre dans lequel les informations apparaissent sur la ligne du fichier journal. Pour plus d'informations sur le schéma XML définissant le format du fichier `FileLoggerFormat.xml`, voir XSD du format de consignateur de fichier autonome.

Types de message

Les agents FTE écrivent une série de types de message différents dans la sous-rubrique SYSTEM.FTE/Log. Pour plus d'informations, voir [SYSTEM.FTE rubrique](#). La définition du fichier journal peut contenir des définitions de format pour les types de message suivants:

```
callCompleted
callStarted
monitorAction
monitorCreate
monitorFired
notAuthorized
scheduleDelete
scheduleExpire
scheduleSkipped
scheduleSubmitInfo
scheduleSubmitTransfer
scheduleSubmitTransferSet
transferStarted
transferCancelled
transferComplete
transferDelete
transferProgress
```

Le format des messages peut varier. La majorité des types de message écrivent une seule ligne dans le fichier journal pour chaque message de journal consommé à partir de la sous-rubrique SYSTEM.FTE/Log. Cela conduit au cas simple où les adresses XPATH fournies dans la définition de format de journal se rapportent à la racine du message. Voici les types de message qui utilisent cette méthode pour écrire la sortie:

```
callCompleted
callStarted
monitorAction
monitorCreate
monitorFired
notAuthorized
scheduleDelete
scheduleExpire
scheduleSkipped
scheduleSubmitInfo
scheduleSubmitTransfer
transferStarted
transferCancelled
transferComplete
transferDelete
```

L'autre méthode utilisée pour écrire un message de journal utilise plusieurs lignes pour représenter les éléments d'un ensemble de transfert dans un message de journal. Dans ce cas, le format fourni est appliqué à chaque élément de l'ensemble de transfert dans le message de journal. Si vous souhaitez inclure des informations spécifiques à chaque élément de l'ensemble de transferts, la valeur XPATH fournie est requise pour utiliser l'élément en tant que racine XPATH. Voici les types de message qui utilisent cette méthode pour écrire la sortie:

```
scheduleSubmitTransferSet
transferProgress
```

Une ligne de sortie est écrite pour chaque élément de l'ensemble de transfert. Les informations que vous souhaitez corriger pour tous les éléments d'un ensemble de transferts peuvent toujours utiliser des adresses XPATH relatives à la racine du message de journal. Dans l'exemple de définition de format `transferProgress` simplifié suivant, il s'agit de l'horodatage et de l'ID de transfert qui sont fixes. Toute information relative à un élément en tant que racine varie pour chaque ligne écrite. Dans cet exemple, les informations de fichier source et de destination de chaque élément sont écrites.

```
<transferProgress>
  <format>
    <inserts>
      <insert type="user" width="19" ignoreNull="false"/>/transaction/action/
```



```

    @time</insert>
    <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
    <insert type="system" width="6" ignoreNull="false">type</insert>
    <insert type="user" width="3" ignoreNull="true">status/@resultCode</insert>
    <insert type="user" width="0" ignoreNull="false">source/file |
    source/queue</insert>
    <insert type="user" width="0" ignoreNull="false">source/file/@size |
    source/queue/@size</insert>
    <insert type="user" width="5" ignoreNull="true">source/@type</insert>
    <insert type="user" width="6" ignoreNull="true">source/@disposition</insert>
    <insert type="user" width="0" ignoreNull="false">destination/file |
    destination/queue</insert>
    <insert type="user" width="0" ignoreNull="false">destination/file/@size |
    destination/queue/@size</insert>
    <insert type="user" width="5" ignoreNull="true">destination/@type</insert>
    <insert type="user" width="9" ignoreNull="true">destination/@exist</insert>
    <insert type="user" width="0" ignoreNull="true">status/supplement</insert>
  </inserts>
  <separator></separator>
</format>
</transferProgress>

```

Cette opération génère une entrée de fichier journal d'une ou de plusieurs lignes au format suivant:

```

2011-11-25T13:45:16;414d5120514d5f67627468696e6b202064666cf4e20033702;[TPRO];0
;/src/test1.file;3575;file;leave ;/dest/test1.file;3575;file;overwrite;;
2011-11-25T13:45:16;414d5120514d5f67627468696e6b202064666cf4e20033702;[TPRO];0
;/src/test2.file;3575;file;leave ;/dest/test2.file;3575;file;overwrite;;

```

Insérer un format

Deux types d'insertion sont disponibles lors de la définition d'un format pour un type de message: **user** et **system**. Le type d'insertion est défini dans l'attribut **type** de l'élément **insert**. Les deux types d'insertion peuvent également avoir leur présentation personnalisée à l'aide des attributs **width** et **ignoreNull** de l'élément **insert**. Exemple :

```

<insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>

```

Dans cet exemple, l'insertion prend les informations trouvées dans le message de journal à l'adresse `/transaction/@ID` et les tronque ou les complète à 48 caractères avant de les écrire dans le journal. Si le contenu de `/transaction/@ID` est null, il écrit la chaîne null après l'avoir remplie à 48 caractères car l'attribut `ignoreNull` est défini sur `false`. Si `ignoreNull` est défini sur `true`, la chaîne vide, remplie à 48 caractères, est écrite à la place. La définition de `width="0"` signifie que la largeur de la colonne n'est pas tronquée, elle ne signifie pas que la largeur est tronquée à 0. L'attribut `ignoreNull` peut être utilisé de cette manière pour détecter dans le journal lorsqu'une valeur null est trouvée alors qu'elle n'était pas attendue. Cela peut être utile lors du débogage d'une nouvelle définition de fichier journal.

Insertions définies par l'utilisateur

Une insertion utilisateur contient une adresse XPATH pour les informations à écrire dans cette insertion. Cette adresse fait référence à une information trouvée dans le message du journal FTE. Pour plus d'informations sur les formats de message de journal, voir:

- [Formats de message de journal de transfert de fichier](#)
- [Formats de message de journal de transfert de fichiers planifiés](#)
- [Format des messages du journal du moniteur MFT](#)

Insertions définies par le système

Les insertions définies par le système contiennent un mot clé qui fait référence à une information qui est introuvable dans le message de journal ou qui n'est pas facile à définir à l'aide du langage XPATH.

Les insertions système prises en charge sont les suivantes:

- `type` -Ecrit le type du message de journal dans un format court.
- `callArguments` -Ecrit l'ensemble des arguments fournis à un appel géré dans un format séparé par un espace.
- `transferMetaData` -Ecrit l'ensemble des entrées de métadonnées définies pour un transfert dans un format `key=value` séparé par des virgules.

Le tableau suivant répertorie la valeur de "type" pour les insertions définies par le système pour chaque type de message.

Tableau 51. Récapitulatif des types de message pris en charge et de leurs insertions système "type".

Type de message	Valeur de l'insertion système "type"
callCompleted	[COM]
callStarted	[CSTR]
monitorAction	[MACT]
monitorCreate	[MCRT]
monitorFired	[MFIR]
notAuthorized	[AUTH]
scheduleDelete	[MODÈLE]
scheduleExpire	[EXP]
scheduleSkipped	[SSKP]
scheduleSubmit-Informations	[SSIN]
Transfert scheduleSubmit	[SSTR]
scheduleSubmitTransferSet	[SSTS]
transferStarted	[TSTR]
transferCancelled	[TCAN]
transferComplete	[TCOM]
transferDelete	[TDEL]
transferProgress	[TPRO]

Référence associée

[Format de journal par défaut du consignateur de fichiers autonome MFT](#)

[Format XSD du consignateur de fichier autonome](#)

[Rubrique SYSTEM.FTE](#)

[Formats des messages du journal de transfert de fichiers](#)

[Formats de message du journal de transfert de fichiers planifiés](#)

[Format des messages du journal du moniteur MFT](#)

ALW**Exclusion de types de message du consignateur de fichiers MFT autonome**

Si vous souhaitez exclure un certain type de message de la sortie du consignateur de fichiers, vous pouvez utiliser des éléments de type de message vides.

Exemple

Par exemple, la définition de format suivante arrête les messages `transferProgress` générés par le consignateur de fichiers.

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="1.00"
  xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <transferProgress></transferProgress>
  </messageTypes>
</logFormatDefinition>
```

ALW**Définition de formats personnalisés pour le consignateur de fichiers MFT autonome**

Il est possible de définir un sous-ensemble de types de message personnalisés dans une définition de format de journal afin de réduire la quantité de configuration requise pour personnaliser le format de votre fichier journal.

Pourquoi et quand exécuter cette tâche

Si un élément `messageTypes` n'est pas inclus dans le fichier `FileLoggerFormat.xml`, le format de ce type de message utilise le format par défaut. Il vous suffit de spécifier les formats que vous souhaitez utiliser comme formats par défaut.

Exemple

Dans cet exemple, la définition de format remplace le format par défaut du type de message `transferStarted` par cette version réduite qui génère uniquement l'utilisateur qui a démarré le transfert. Tous les autres types de message utilisent le format par défaut car ils ne sont pas inclus dans cette définition de format de journal:

```
<?xml version="1.0" encoding="UTF-8"?>
<logFormatDefinition xmlns:xsi="https://www.w3.org/2001/XMLSchema-instance" version="1.00"
  xsi:noNamespaceSchemaLocation="FileLoggerFormat.xsd">
  <messageTypes>
    <transferStarted>
      <format>
        <inserts>
          <insert type="user" width="19" ignoreNull="false">/transaction/action/
            @time</insert>
          <insert type="user" width="48" ignoreNull="false">/transaction/@ID</insert>
          <insert type="system" width="6" ignoreNull="false">type</insert>
          <insert type="user" width="0" ignoreNull="true">/transaction/originator/
            userID</insert>
        </inserts>
        <separator>;</separator>
      </format>
    </transferStarted>
  </messageTypes>
</logFormatDefinition>
```

Référence associée

[Format de journal par défaut du consignateur de fichiers autonome MFT](#)

[Format XSD du consignateur de fichier autonome](#)

ALW**Réduction des messages en double dans le consignateur de fichiers MFT autonome**

Des messages de journal en double peuvent apparaître dans le journal du consignateur de fichiers autonome. À l'aide du fichier `logger.properties`, vous pouvez optimiser le consignateur de fichiers autonome et réduire le nombre de doublons.

Messages en double dans le journal du consignateur de fichiers

En cas d'échec, un message de journal peut être enregistré dans le journal du consignateur de fichiers autonome sans que le message de journal provenant de SYSTEM.FTE/Log# en cours de validation dans IBM MQ. Si cela se produit, lorsque le consignateur de fichiers autonome redémarre, il récupère le même message une deuxième fois et l'écrit à nouveau dans le fichier journal. Prévoyez de gérer la possibilité de ces doublons lorsque vous consultez les fichiers journaux manuellement ou lors de leur traitement automatique. Pour faciliter la détection des doublons, le consignateur de fichiers autonomes génère le message suivant dans le fichier journal lorsqu'il démarre:

```
BFGDB0054I: The file logger has successfully started
```

Les doublons se produisent toujours autour de l'heure de début du consignateur de fichiers autonome, car c'est lorsque le dernier message lu avant l'échec de l'instance précédente est traité. En sachant quand la nouvelle instance a démarré, vous pouvez détecter si des doublons doivent être attendus et s'ils doivent être traités ou non.

Réduction du nombre de doublons

Le consignateur de fichiers autonome regroupe les messages de journal qu'il traite dans des transactions afin d'améliorer les performances. Cette taille de lot correspond au nombre maximal de messages en double que vous pouvez voir en cas d'échec. Pour réduire le nombre de doublons, vous pouvez optimiser la propriété suivante dans le fichier `logger.properties` :

```
wmqfte.max.transaction.messages
```

Par exemple, en définissant cette valeur sur 1, le nombre maximal de messages dupliqués est réduit à 1. Sachez que la modification de cette valeur a un impact sur les performances de votre consignateur de fichiers autonome. Par conséquent, des tests approfondis sont nécessaires pour s'assurer que cela n'affecte pas votre système.

Le fichier `logger.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`. Pour plus d'informations sur les propriétés disponibles et leurs effets, voir [Propriétés de configuration du consignateurMFT](#)

Configuration de l'accès utilisateur pour un consignateur de fichiers MFT autonome

Dans un environnement de test, vous pouvez ajouter tous les nouveaux privilèges nécessaires à votre compte utilisateur normal. Dans un environnement de production, il est recommandé de créer un utilisateur avec les droits minimum requis pour effectuer le travail.

Pourquoi et quand exécuter cette tâche

Vous devez installer le consignateur de fichiers autonome et IBM MQ sur un système unique. Configurez les droits de l'utilisateur comme suit:

Procédure

1. Vérifiez que l'utilisateur est autorisé à lire et, si nécessaire, à exécuter les fichiers installés dans le cadre de l'installation de Managed File Transfer .
2. Vérifiez que l'utilisateur est autorisé à créer et à écrire dans n'importe quel fichier du répertoire `logs` qui se trouve dans le répertoire de configuration. Ce répertoire est utilisé pour un journal des événements et, si nécessaire, pour les fichiers de trace de diagnostic et FFDC (First Failure Data Capture).
3. Assurez-vous que l'utilisateur possède son propre groupe et qu'il ne fait partie d'aucun groupe disposant de droits étendus sur le gestionnaire de files d'attente de coordination. L'utilisateur ne doit pas faire partie du groupe `mqm`. Sur certaines plateformes, le groupe de personnel reçoit

également automatiquement l'accès au gestionnaire de files d'attente ; l'utilisateur du consigneur de fichiers autonome ne doit pas se trouver dans le groupe de personnel. Vous pouvez afficher les enregistrements de droits d'accès pour le gestionnaire de files d'attente lui-même et pour les objets qu'il contient à l'aide de la IBM MQ Explorer. Cliquez avec le bouton droit de la souris sur l'objet et sélectionnez **Droits sur les objets > Gérer les enregistrements de droits d'accès**. Sur la ligne de commande, vous pouvez utiliser les commandes `dspmqaut` (display authority) ou `dmpmqaut` (dump authority).

4. Utilisez la fenêtre **Gérer les enregistrements de droits d'accès** dans IBM MQ Explorer ou la commande `setmqaut` (grant or revoke authority) pour ajouter des droits d'accès au groupe de l'utilisateur (sur AIX, les droits d'accès IBM MQ sont associés à des groupes uniquement, et non à des utilisateurs individuels). Les autorités requises sont les suivantes:

- Connexion et interrogation sur le gestionnaire de files d'attente (les bibliothèques IBM MQ Java requièrent des droits d'interrogation pour fonctionner).
- Droit d'abonnement sur SYSTEM.FTE .
- Placez les droits sur le système SYSTEM.FTE.LOG.RJCT.*nom_journal* file d'attente.
- Obtenez les droits sur SYSTEM.FTE.LOG.CMD.*nom_journal* file d'attente.

Les noms de file d'attente de rejet et de commandes indiqués sont les noms par défaut. Si vous avez choisi des noms de file d'attente différents lors de la configuration des files d'attente du consigneur de fichiers autonomes, ajoutez les droits d'accès à ces noms de file d'attente à la place.

Installation du consigneur de base de données autonome MFT

Procédez comme suit pour installer et configurer le consigneur de base de données autonome.

Pourquoi et quand exécuter cette tâche


Important : Les programmes de consignation Managed File Transfer ne sont pas pris en charge sur la plateforme IBM i .

Pour plus d'informations sur le consigneur de base de données autonome, voir [«Configuration d'un consigneur MFT»](#), à la page 852.

Remarque : Vous ne pouvez pas exécuter simultanément plusieurs consigneurs de base de données (autonomes ou Java EE) sur le même schéma dans une base de données. Une telle tentative entraînerait des conflits lors de la tentative d'écriture des données de journal de transfert dans la base de données.

Procédure

1. Installez votre logiciel de base de données à l'aide de la documentation de votre base de données.
Si la prise en charge de JDBC est un composant facultatif pour votre base de données, vous devez installer ce composant.
2. Exécutez la commande **fteCreateLogger** en définissant le paramètre **-loggerType** sur DATABASE pour créer votre consigneur de base de données autonome. Pour plus d'informations, voir [fteCreateLogger](#).
Le nom de schéma par défaut est FTELOG. Si vous utilisez un nom de schéma autre que FTELOG, vous devez éditer le fichier SQL fourni correspondant à votre base de données, `ftelog_tables_db2.sql` ou `ftelog_tables_oracle.sql`, pour refléter ce nom de schéma avant de passer à l'étape suivante. Pour plus d'informations, voir `wmqfte.database.schema` dans les propriétés de configuration du consigneur [MFT](#).
3. Créez les tables de base de données requises à l'aide des outils de votre base de données.

 Sous [Multiplateformes](#), les fichiers `ftelog_tables_db2.sql` et `ftelog_tables_oracle.sql` contiennent des commandes SQL que vous pouvez exécuter pour créer les tables.

z/OS

Sous z/OS, le fichier que vous devez exécuter dépend de la version de Db2 for z/OS que vous utilisez:

- Pour Db2 for z/OS 9.0 et les versions antérieures, exécutez le fichier `ftelog_tables_zos.sql` pour créer les tables. Ce fichier crée les tables à l'aide d'un type de données INTEGER pour les zones qui indiquent les tailles des fichiers transférés et l'ID de table associé à chaque transfert.
 - Pour Db2 for z/OS 9.1 et les versions ultérieures, exécutez le fichier `ftelog_tables_zos_bigint.sql` pour créer les tables. Ce fichier crée les tables à l'aide d'un type de données BIGINT pour les zones qui indiquent les tailles des fichiers transférés et l'ID de table associé à chaque transfert.
4. Exécutez les commandes MQSC, fournies par la commande **fteCreateLogger**, sur votre gestionnaire de files d'attente de commandes de consignateur pour créer les files d'attente de consignateur. Le consignateur de base de données autonome utilise deux files d'attente sur le gestionnaire de file d'attente de coordination. La première file d'attente est une file d'attente de commandes dans laquelle sont placés les messages permettant de contrôler le fonctionnement du consignateur de base de données autonome. Le nom par défaut de cette file d'attente de commandes est `SYSTEM.FTE.LOG.CMD.nom_journal`. La deuxième file d'attente est une file d'attente de rejet. Etant donné que le consignateur de base de données autonome ne supprime jamais les messages de journal, si le consignateur détecte un message qu'il ne peut pas traiter, il place le message dans la file d'attente de rejet pour examen et le retraite éventuellement. Il n'est pas recommandé d'utiliser la file d'attente de rebut du gestionnaire de files d'attente à cette fin, car les messages rejetés n'ont pas d'en-tête DLH et les messages rejetés ne doivent pas être combinés avec les messages placés dans la file d'attente de rebut pour d'autres raisons. Le nom par défaut de la file d'attente de rejet est `SYSTEM.FTE.LOG.RJCT.nom_journal`. Ces deux files d'attente sont définies dans les fichiers script MQSC générés par la commande **fteCreateLogger**.
 5. Choisir un utilisateur et configurer les droits d'accès
 6. Facultatif : Vous pouvez configurer davantage le consignateur de base de données autonome en éditant le fichier `logger.properties` créé par la commande **fteCreateLogger** à l'étape «2», à la page 861. Ce fichier est un fichier de propriétés Java qui se compose de paires clé-valeur. Le fichier `logger.properties` se trouve dans le répertoire `MQ_DATA_PATH/mqft/config/coordination_qmgr_name/loggers/logger_name`. Pour plus d'informations sur les propriétés disponibles et leurs effets, voir Propriétés de configuration du consignateurMFT.
 7. **Windows**
Facultatif : Si vous utilisez un système Windows, vous pouvez exécuter le consignateur de base de données autonome en tant que service Windows. Exécutez la commande **fteModifyLogger** avec le paramètre **-s**. Pour plus d'informations, voir fteModifyLogger.
 8. Facultatif : Si la base de données utilisée est Oracle ou si vous vous connectez à une base de données Db2 à distance, vous devez spécifier un nom d'utilisateur et un mot de passe que le consignateur utilisera pour s'authentifier auprès de votre serveur de base de données. Ce nom d'utilisateur et ce mot de passe sont spécifiés dans un fichier de données d'identification conforme au format défini par le schéma `MQMFTCcredentials.xsd`. Pour plus d'informations, voir Format de fichier des données d'identification MFT. Après avoir créé le fichier de données d'identification, vous devez spécifier l'emplacement du fichier de données d'identification dans le fichier `logger.properties` à l'aide de la propriété `wmqfte.database.credentials.file`.
 9. Démarrez le consignateur de base de données autonome à l'aide de la commande **fteStartLogger**. Par défaut, le consignateur de base de données autonome s'exécute en arrière-plan et le consignateur de base de données autonome place la sortie dans un fichier du répertoire `logs`. Si vous souhaitez exécuter le consignateur de base de données autonome en avant-plan et générer une sortie sur la console ainsi que dans le fichier journal, ajoutez le paramètre **-F** à la commande **fteStartLogger**.

Si vous avez effectué l'étape précédente et utilisé la commande **fteModifyLogger** avec le paramètre **-s** sous Windows, le consignateur de base de données autonome démarre en tant que service Windows.

Tâches associées

«Configuration de l'accès utilisateur pour un consignateur de base de données autonome MFT», à la page 863

Dans un environnement de test, vous pouvez ajouter tous les nouveaux privilèges nécessaires à votre compte utilisateur normal. Dans un environnement de production, il est recommandé de créer un utilisateur avec les droits minimum requis pour effectuer le travail.

Référence associée

[Propriétés de configuration du consignateur MFT](#)

[fteStartLogger](#)

[fteModifyLogger](#)

[Autorisations pour le consignateur MFT](#)

Utilisation de MFT avec une base de données distante

Vous pouvez utiliser le consignateur Managed File Transfer pour communiquer avec une base de données sur un système distant.

Pourquoi et quand exécuter cette tâche

Si une base de données est installée sur une machine différente de celle sur laquelle Managed File Transfer est installé, procédez comme suit. Sauf indication contraire, les étapes s'appliquent à Db2 et à Oracle .

Procédure

1. Installez un client de base de données sur le système sur lequel vous avez installé Managed File Transfer .
2. Ajoutez votre serveur de base de données distant à votre configuration de client de base de données locale. Cette mise à jour de la configuration est nécessaire pour que Managed File Transfer et IBM MQ puissent accéder correctement à la base de données.
3. Spécifiez les nouvelles propriétés dans le fichier `logger.properties` pour vous connecter à la base de données à l'aide du fichier de données d'identification: **`wmfte.database.credentials.file`**.

Remarque : Deprecated Les versions antérieures de Managed File Transfer utilisaient les propriétés **`wmqfte.oracle.user`** ou **`wmqfte.database.user`** et **`wmqfte.oracle.password`** ou **`wmqfte.database.password`**. Ces propriétés sont désormais obsolètes. Utilisez **`wmfte.database.credentials.file`** à la place.

4. **Oracle uniquement:** Pour autoriser une connexion distante à la base de données, modifiez la strophe `XAResourceManager` dans le fichier `qm.ini` du gestionnaire de file d'attente de coordination comme suit (en vous assurant que vous modifiez le nom de la base de données, le nom d'utilisateur et le mot de passe de l'utilisateur pour qu'ils correspondent à vos propres informations):

```
Oracle_XA+Acc=P/fte/ log/  
qgw783jhT+SesTm=35+DB=FTEAUDIT1+SqlNet=FTEAUDIT1+threads=false,  
la modification est mise en évidence en gras.
```

5. **Oracle uniquement:** Indiquez un hôte et un port dans le fichier `logger.properties`, à l'aide des propriétés **`wmqfte.oracle.host`** et **`wmqfte.oracle.port`**. Les valeurs par défaut de l'hôte et du port vous permettent d'utiliser un client de base de données local. Par conséquent, si vous avez déjà utilisé une base de données locale, vous n'avez peut-être pas défini ces valeurs.

Référence associée

[Propriétés de configuration du consignateur MFT](#)

Configuration de l'accès utilisateur pour un consignateur de base de données autonome MFT

Dans un environnement de test, vous pouvez ajouter tous les nouveaux privilèges nécessaires à votre compte utilisateur normal. Dans un environnement de production, il est recommandé de créer un utilisateur avec les droits minimum requis pour effectuer le travail.

Pourquoi et quand exécuter cette tâche

Le nombre et le type de comptes utilisateur dont vous avez besoin pour exécuter le consigneur de base de données autonome dépendent du nombre de systèmes que vous utilisez. Vous pouvez installer le consigneur de base de données autonome, IBM MQ et votre base de données sur un seul système ou sur deux systèmes. Le consigneur de base de données autonome doit se trouver sur le même système que IBM MQ. Les composants peuvent être installés dans les topologies suivantes:

Consigneur de base de données autonome, IBM MQ et la base de données se trouvent tous sur le même système

Vous pouvez définir un utilisateur de système d'exploitation unique à utiliser avec les trois composants. Il s'agit d'une configuration appropriée pour le consigneur de base de données autonome. Le consigneur de base de données autonome utilise le mode Liaisons pour se connecter à IBM MQ et une connexion native pour se connecter à la base de données.

Consigneur de base de données autonome et IBM MQ sur un système, la base de données sur un système distinct

Vous créez deux utilisateurs pour cette configuration: un utilisateur du système d'exploitation sur le système exécutant le consigneur de base de données autonome et un utilisateur du système d'exploitation disposant d'un accès distant à la base de données sur le serveur de base de données. Il s'agit d'une configuration appropriée pour le consigneur de base de données autonome utilisant une base de données distante. Le consigneur de base de données autonome utilise le mode Liaisons pour se connecter à IBM MQ et une connexion client pour accéder à la base de données.

Par exemple, le reste de ces instructions suppose que l'utilisateur est appelé `fteLog`, mais vous pouvez utiliser n'importe quel nom d'utilisateur. Configurez les droits de l'utilisateur comme suit:

Procédure

1. Vérifiez que l'utilisateur est autorisé à lire et, si nécessaire, à exécuter les fichiers installés dans le cadre de l'installation de Managed File Transfer Remote Tools and Documentation.
2. Vérifiez que l'utilisateur est autorisé à créer et à écrire dans n'importe quel fichier du répertoire `logs` (dans le répertoire de configuration). Ce répertoire est utilisé pour un journal des événements et, si nécessaire, pour les fichiers de trace de diagnostic et FFDC.
3. Vérifiez que l'utilisateur possède son propre groupe et qu'il ne fait partie d'aucun groupe disposant de droits étendus sur le gestionnaire de files d'attente de coordination. L'utilisateur ne doit pas faire partie du groupe `mqm`. Sur certaines plateformes, le groupe de personnel reçoit également automatiquement l'accès au gestionnaire de files d'attente ; l'utilisateur du consigneur de base de données autonome ne doit pas figurer dans le groupe de personnel. Vous pouvez afficher les enregistrements de droits d'accès pour le gestionnaire de files d'attente lui-même et pour les objets qu'il contient à l'aide de la IBM MQ Explorer. Cliquez avec le bouton droit de la souris sur l'objet et sélectionnez **Droits sur les objets > Gérer les enregistrements de droits d'accès**. Sur la ligne de commande, vous pouvez utiliser les commandes `dspmqaqt (display authority)` ou `dmpmqaqt (dump authority)`.
4. Utilisez la fenêtre **Gérer les enregistrements de droits d'accès** dans IBM MQ Explorer ou la commande `setmqaut (grant or revoke authority)` pour ajouter des droits d'accès au groupe de l'utilisateur (sur AIX, les droits d'accès IBM MQ sont associés à des groupes uniquement, et non à des utilisateurs individuels). Les autorités requises sont les suivantes:
 - Connexion et interrogation sur le gestionnaire de files d'attente (les bibliothèques IBM MQ Java requièrent des droits d'interrogation pour fonctionner).
 - Droit d'abonnement sur `SYSTEM.FTE` .
 - Placez les droits sur le système `SYSTEM.FTE.LOG.RJCT.nom_journal` file d'attente.
 - Obtenez les droits sur `SYSTEM.FTE.LOG.CMD.nom_journal` file d'attente.Les noms de file d'attente de rejet et de commandes indiqués sont les noms par défaut. Si vous avez choisi des noms de file d'attente différents lors de la configuration des files d'attente du consigneur de base de données autonome, ajoutez les droits à ces noms de file d'attente à la place.
5. Effectuez la configuration utilisateur spécifique à la base de données que vous utilisez.

- Si votre base de données est Db2, procédez comme suit:

Il existe plusieurs mécanismes de gestion des utilisateurs de base de données avec Db2. Ces instructions s'appliquent au schéma par défaut basé sur les utilisateurs du système d'exploitation.

- Vérifiez que l'utilisateur `fte1log` ne fait partie d'aucun groupe d'administration Db2 (par exemple, `db2iadm1`, `db2fadm1` ou `dasadm1`)
- Accordez à l'utilisateur le droit de se connecter à la base de données et le droit de sélectionner, d'insérer et de mettre à jour les tables que vous avez créées dans le cadre de l' [étape 2: création des tables de base de données requises](#)

- Si votre base de données est Oracle, procédez comme suit:

- Vérifiez que l'utilisateur `fte1log` ne fait partie d'aucun groupe d'administration Oracle (par exemple, `ora_dba` sous Windows ou `dba` sous AIX and Linux)
- Accordez à l'utilisateur le droit de se connecter à la base de données et le droit de sélectionner, d'insérer et de mettre à jour les tables que vous avez créées dans le cadre de l' [étape 2: création des tables de base de données requises](#)

Autres configurations pour un consignateur MFT autonome

En règle générale, un consignateur Managed File Transfer autonome, qu'il s'agisse d'un fichier ou d'un type de base de données, se trouve sur le même système que le gestionnaire de files d'attente de coordination et est connecté au gestionnaire de files d'attente de coordination en mode liaisons IBM MQ. Toutefois, il peut également être installé sur le même système que n'importe quel gestionnaire de files d'attente disposant d'une connectivité au gestionnaire de files d'attente de coordination. Le consignateur autonome reçoit des messages à l'aide d'un abonnement, que le consignateur autonome crée automatiquement. Il s'agit de la configuration décrite dans les instructions d'installation.

Toutefois, si vous avez des considérations spécifiques au site, vous pouvez configurer un consignateur autonome pour qu'il reçoive les messages de deux autres manières, contrôlées par la propriété `wmqfte.message.source.type`. Cette propriété est décrite dans [Propriétés de configuration du consignateurMFT](#).

Abonnement d'administration

Par défaut, un consignateur autonome crée son propre abonnement à `SYSTEM.FTE/Log/#`, utilisation des options d'abonnement durable par défaut et d'un abonnement géré (c'est-à-dire que le gestionnaire de files d'attente contrôle la file d'attente de sauvegarde utilisée pour stocker les messages avant qu'ils ne soient transmis à l'application). Si d'autres options sont requises sur l'abonnement ou la file d'attente, vous pouvez créer un abonnement vous-même, définir les options dont vous avez besoin et configurer le consignateur autonome pour qu'il utilise cet abonnement à la place. N'oubliez pas d'ajouter le droit permettant au consignateur autonome d'utiliser l'abonnement que vous créez.

Un exemple d'utilisation de cette configuration consiste à partitionner l'espace de journal à l'aide de deux abonnements génériques, à envoyer des journaux à partir d'agents dont le nom commence par `FINANCE` dans une base de données et des journaux à partir d'agents commençant par `ACCOUNTING` dans une autre. Ce type de configuration requiert deux instances de consignateur autonomes, chacune avec son propre fichier `logger.properties` faisant référence à l'abonnement requis et sa propre file d'attente de commandes et de rejets.

Pour collecter des messages de journal uniquement à partir d'agents dont le nom commence par `ACCOUNTING`, créez un objet d'abonnement sur votre gestionnaire de files d'attente de coordination avec une chaîne de rubrique `SYSTEM.FTE/Log/ACCOUNTING*`. Définissez la valeur **Utilisation des caractères génériques** sur **Caractère générique de niveau caractère**. Vous devez également ajouter des entrées au fichier `logger.properties` pour votre consignateur. Par exemple, si vous créez un objet

d'abonnement appelé ACCOUNTING.LOGS avec ces paramètres, ajoutez les entrées suivantes au fichier `logger.properties` :

```
wmqfte.message.source.type=administrative subscription
wmqfte.message.source.name=ACCOUNTING.LOGS
```

Le consignateur autonome gère les messages de journal qui commencent par la chaîne de rubrique SYSTEM.FTE/Log/ uniquement. Vous pouvez spécifier une chaîne de rubrique plus restrictive, mais pas une chaîne moins restrictive. Si vous spécifiez une chaîne d'erreur moins restrictive, toutes les publications associées à une chaîne de rubrique autre que SYSTEM.FTE/Log/ vont dans la file d'attente de rejet et le consignateur autonome génère le message d'erreur BFGDB0002E. Ce message d'erreur implique un problème avec la configuration du consignateur autonome.

File d'attente

La topologie typique est celle où le consignateur autonome s'exécute sur le même système que le gestionnaire de files d'attente de coordination. Si cela n'est pas possible, vous pouvez créer un abonnement sur le gestionnaire de files d'attente de coordination en utilisant une file d'attente sur un autre gestionnaire de files d'attente comme destination d'abonnement (à l'aide d'une définition de file d'attente éloignée ou de la propriété DESTQMGR de l'abonnement). Le consignateur peut ensuite s'exécuter sur le système hébergeant le deuxième gestionnaire de files d'attente et lire les messages de la file d'attente. Pour garantir l'intégrité transactionnelle, le consignateur autonome doit toujours se connecter à son gestionnaire de files d'attente en mode liaisons. Vous devez définir la file d'attente de rejet et la file d'attente de commandes sur le même gestionnaire de files d'attente auquel le consignateur autonome se connecte. Les gestionnaires de files d'attente doivent être au niveau IBM WebSphere MQ 7.5 ou ultérieur.

Par exemple, pour collecter des messages de journal qui sont placés dans la file d'attente USER.QUEUE par un abonnement, ajoutez ces entrées au fichier `logger.properties` :

```
wmqfte.message.source.type=queue
wmqfte.message.source.name=USER.QUEUE
```

Installation du consignateur de base de données Java EE pour MFT

Suivez ces instructions pour installer et configurer le consignateur de base de données JEE à utiliser avec Managed File Transfer.

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur le consignateur de base de données Java EE , voir la rubrique [«Configuration d'un consignateur MFT»](#), à la page 852.

Remarque : Vous ne pouvez pas exécuter un consignateur de base de données Java EE en même temps qu'un consignateur autonome, sauf si ces consignateurs utilisent des instances distinctes de la base de données.

Procédure

1. Avant d'installer le consignateur de base de données Java EE , vous devez préparer votre environnement. Utilisez les instructions de la rubrique [«Préparation de l'installation du consignateur de base de données Java EE pour MFT»](#), à la page 867.
2. Installez le consignateur de base de données Java EE dans un serveur d'applications Java Platform, Enterprise Edition (Java EE) ou Jakarta EE compatible.

Pour des instructions, reportez-vous à l' [«Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server traditional 9.0»](#), à la page 870

Tâches associées

«Préparation de l'installation du consigneur de base de données Java EE pour MFT», à la page 867
Suivez ces instructions pour préparer votre environnement Managed File Transfer avant d'installer le consigneur de base de données Java EE .

«Installation du consigneur de base de données Java EE pour MFT avec WebSphere Application Server traditional 9.0», à la page 870

Suivez ces instructions pour installer et configurer le consigneur de base de données Java Platform, Enterprise Edition (Java EE) pour Managed File Transfer avec WebSphere Application Server traditional 9.0.

«Configuration de l'accès utilisateur pour le consigneur de base de données Java EE pour MFT», à la page 875

Lorsque vous configurez le consigneur de base de données Java Platform, Enterprise Edition (Java EE) pour Managed File Transfer, vous avez besoin de comptes utilisateur pour accéder à IBM MQ, à votre base de données et à votre système d'exploitation. Le nombre d'utilisateurs du système d'exploitation requis dépend du nombre de systèmes que vous utilisez pour héberger ces composants.

«Migration du consigneur de base de données autonome vers le consigneur de base de données Java EE pour MFT», à la page 876

Vous pouvez effectuer une migration depuis le consigneur de base de données autonome vers le consigneur de base de données Java EE . Vous devez arrêter le consigneur de base de données autonome et installer le consigneur de base de données JEE. Pour éviter de perdre ou de dupliquer des entrées de journal, vous devez arrêter la publication de messages dans SYSTEM.FTE avant d'arrêter le consigneur de base de données autonome et de le redémarrer après avoir installé le consigneur de base de données Java EE . Sauvegardez votre base de données avant la migration.

Référence associée

[Autorisations pour le consigneur MFT](#)

Préparation de l'installation du consigneur de base de données Java EE pour MFT

Suivez ces instructions pour préparer votre environnement Managed File Transfer avant d'installer le consigneur de base de données Java EE .

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur le consigneur de base de données Java EE , voir la rubrique «[Configuration d'un consigneur MFT](#)», à la page 852.

Procédure

1. Installez votre logiciel de base de données à l'aide de la documentation de votre base de données.
Si la prise en charge de JDBC est un composant facultatif pour votre base de données, vous devez installer ce composant.
2. Créez une base de données à l'aide des outils fournis par votre base de données. La base de données doit disposer d'un espace table et d'une taille de page de pool de mémoire tampon d'au moins 8K.
Le nom de schéma par défaut est FTELOG. Si vous utilisez un nom de schéma autre que FTELOG, vous devez éditer le fichier SQL fourni correspondant à votre base de données, `ftelog_tables_db2.sql` ou `ftelog_tables_oracle.sql`, pour le refléter avant de passer à l'étape suivante.

Remarque : Les fichiers `ftelog_tables_db2.sql` et `ftelog_tables_oracle.sql` se trouvent dans le chemin d'accès au fichier `<MQ-installation-path>/mqft/sql`

3. Créez les tables de base de données requises à l'aide des outils de votre base de données.

Multi Sous [Multiplateformes](#), les fichiers `ftelog_tables_db2.sql` et `ftelog_tables_oracle.sql` contiennent des commandes SQL que vous pouvez exécuter pour créer les tables.

z/OS Sous z/OS, le fichier que vous devez exécuter dépend de la version de Db2 for z/OS que vous utilisez:

- Pour Db2 for z/OS 9.0 et les versions antérieures, exécutez le fichier `fteelog_tables_zos.sql` pour créer les tables. Ce fichier crée les tables à l'aide d'un type de données INTEGER pour les zones qui indiquent les tailles des fichiers transférés et l'ID de table associé à chaque transfert.
 - Pour Db2 for z/OS 9.1 et les versions ultérieures, exécutez le fichier `fteelog_tables_zos_bigint.sql` pour créer les tables. Ce fichier crée les tables à l'aide d'un type de données BIGINT pour les zones qui indiquent les tailles des fichiers transférés et l'ID de table associé à chaque transfert.
4. Si vous avez modifié le nom de schéma FTELOG, vous devez modifier le nom de schéma dans le fichier EAR. Pour plus d'informations, voir [«Modification du nom de schéma dans le consignateur de base de données Java EE pour MFT»](#), à la page 868.
 5. Créez une file d'attente de rejet dans IBM MQ.
Etant donné que le consignateur ne supprime jamais les messages de journal, si le consignateur rencontre un message qu'il ne peut pas traiter, il place le message dans la file d'attente de rejet pour examen et traitement éventuel. N'utilisez pas la file d'attente de rebut du gestionnaire de files d'attente à cette fin, car les messages rejetés n'ont pas d'en-tête DLH et parce que les messages rejetés ne doivent pas être combinés avec les messages placés dans la file d'attente de rebut pour d'autres raisons. La commande **fteCreateLogger** crée une file d'attente de rejet. Le nom par défaut de cette file d'attente de rejet est `SYSTEM.FTE.LOG.RJCT.nom_connexion`
 6. Suivez les instructions de la rubrique [«Configuration de l'accès utilisateur pour le consignateur de base de données Java EE pour MFT»](#), à la page 875.

Que faire ensuite

Installez le consignateur de base de données Java EE sur un serveur d'applications compatible Java EE ou Jakarta EE . Utilisez les instructions de la rubrique [«Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server traditional 9.0»](#), à la page 870

Modification du nom de schéma dans le consignateur de base de données Java EE pour MFT

Le consignateur de base de données Java Platform, Enterprise Edition (Java EE) peut utiliser une base de données dont le nom de schéma n'est pas celui par défaut. Vous devez modifier le nom de schéma dans le fichier EAR du consignateur de base de données Java EE .

Pourquoi et quand exécuter cette tâche

Pour modifier le nom du schéma utilisé par le consignateur de base de données Java EE , procédez comme suit:

Procédure

1. Extrayez le fichier JAR JPA du fichier EAR à l'aide de la commande suivante:

```
jar -xvf ear_file lib/jpa_file
```

où :

- `ear_file` est `com.ibm.wmqfte.databaselogger.jee.oracle.ear` ou `com.ibm.wmqfte.databaselogger.jee.ear` selon que vous utilisez Db2 ou Oracle.
- `jpa_file` est `com.ibm.wmqfte.web.jpa.oracle.jar` ou `com.ibm.wmqfte.web.jpa.jar` selon que vous utilisez Db2 ou Oracle.

2. Extrayez le fichier `persistence.xml` du fichier JAR JPA à l'aide de la commande suivante:

```
jar -xvf lib/jpa_file META_INF/persistence.xml
```

où :

- *jpa_file* est `com.ibm.wmqfte.web.jpa.oracle.jar` ou `com.ibm.wmqfte.web.jpa.jar` selon que vous utilisez Db2 ou Oracle.

3. Editez le fichier `persistance.xml` pour modifier la ligne suivante:

```
<property name="openjpa.jdbc.Schema" value="schema_name" />
```

Où

- *nom_schéma* est le nom de schéma que vous souhaitez utiliser.

4. Mettez à jour le fichier JAR JPA avec le fichier `persistance.xml` modifié à l'aide de la commande suivante:

```
jar -uvf lib/jpa_file META_INF/persistance.xml
```

où :

- *jpa_file* est `com.ibm.wmqfte.web.jpa.oracle.jar` ou `com.ibm.wmqfte.web.jpa.jar` selon que vous utilisez Db2 ou Oracle.

5. Mettez à jour le fichier EAR avec le fichier JAR JPA modifié à l'aide de la commande suivante:

```
jar -uvf ear_file lib/jpa_file
```

où :

- *ear_file* est `com.ibm.wmqfte.databaselogger.jee.oracle.ear` ou `com.ibm.wmqfte.databaselogger.jee.ear` selon que vous utilisez Db2 ou Oracle.
- *jpa_file* est `com.ibm.wmqfte.web.jpa.oracle.jar` ou `com.ibm.wmqfte.web.jpa.jar` selon que vous utilisez Db2 ou Oracle.

Que faire ensuite

Utilisez le fichier EAR modifié pour installer le consignateur de base de données Java EE .

Tâches associées

«[Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server traditional 9.0](#)», à la page 870

Suivez ces instructions pour installer et configurer le consignateur de base de données Java Platform, Enterprise Edition (Java EE) pour Managed File Transfer avec WebSphere Application Server traditional 9.0.

Définition du chemin d'accès à la bibliothèque native dans WebSphere Application Server traditional 9.0

Si vous déployez l'application de consignateur de base de données Java Platform, Enterprise Edition (Java EE) sous WebSphere Application Server traditional 9.0 et que vous souhaitez utiliser des connexions en mode liaisons entre l'application et IBM MQ, vous devez configurer le fournisseur de messagerie IBM MQ avec l'emplacement des bibliothèques natives IBM MQ sur le système.

Pourquoi et quand exécuter cette tâche

Si vous ne définissez pas le chemin d'accès à la bibliothèque native dans votre serveur d'applications, vous risquez de recevoir le message d'erreur suivant dans le journal de sortie du système WebSphere Application Server traditional 9.0 :

```
A connection could not be made to WebSphere MQ for the following reason:  
CC=2;RC=2495;AMQ8568: The native JNI library 'mqjbnd' was not found. [3=mqjbnd]
```

Utilisez la console d'administration WebSphere Application Server traditional 9.0 pour effectuer les étapes suivantes:

Procédure

1. Dans le panneau de navigation, développez **Ressources** > **JMS** > **Fournisseurs JMS**.
2. Sélectionnez le fournisseur de messagerie IBM MQ dont la portée est correcte pour la fabrique de connexions ou la spécification d'activation qui crée la connexion en mode liaisons.

Remarque : Les informations de chemin natif dans la portée `Server` sont utilisées de préférence aux informations de chemin natif dans les portées supérieures, et les informations de chemin natif dans la portée `Node` sont utilisées de préférence aux informations de chemin natif dans la portée `Cell` .

3. Sous Propriétés générales, dans la zone **Chemin d'accès à la bibliothèque native** , entrez le nom complet du répertoire contenant les bibliothèques natives IBM MQ .

Par exemple, sous Linux , entrez `/opt/mqm/java/lib`. Entrez un seul nom de répertoire.

4. Cliquez sur **OK**.

Une fois le chemin défini, vous devez enregistrer les modifications dans la configuration principale pour que les modifications soient prises en compte.

5. Redémarrez le serveur d'applications pour actualiser la configuration.
6. Obligatoire : Redémarrez le serveur d'applications une deuxième fois pour charger les bibliothèques.

Tâches associées

«Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server traditional 9.0», à la page 870

Suivez ces instructions pour installer et configurer le consignateur de base de données Java Platform, Enterprise Edition (Java EE) pour Managed File Transfer avec WebSphere Application Server traditional 9.0.

Installation du consignateur de base de données Java EE pour MFT avec WebSphere Application Server traditional 9.0

Suivez ces instructions pour installer et configurer le consignateur de base de données Java Platform, Enterprise Edition (Java EE) pour Managed File Transfer avec WebSphere Application Server traditional 9.0.

Avant de commencer


Avant d'installer l'application de consignateur de base de données JEE, suivez les instructions des rubriques «Préparation de l'installation du consignateur de base de données Java EE pour MFT», à la page 867 et «Définition du chemin d'accès à la bibliothèque native dans WebSphere Application Server traditional 9.0», à la page 869.

Pourquoi et quand exécuter cette tâche

Pour plus d'informations sur le consignateur de base de données Java EE , voir «Configuration d'un consignateur MFT», à la page 852.

Procédure

1. Configurez le fournisseur JDBC XA:
 - a) Sélectionnez **Ressources** > **JDBC** > **JDBC Fournisseurs** dans la navigation de la console d'administration WebSphere Application Server traditional 9.0 .
 - b) Créez un fournisseur JDBC à l'aide de l'assistant de console en cliquant sur **Nouveau**.
 - c) A l'étape 1 de l'assistant, sélectionnez la base de données que vous utilisez dans la liste **Type de base de données** et le type de fournisseur associé dans la liste **Type de fournisseur** . Dans la liste **Type d'implémentation** , sélectionnez **Source de données XA**. Cliquez sur **Suivant**.

 Vous pouvez supprimer une référence à `db2jcc_license_cisuz.jar` et remplacer `db2jcc.jar` par `db2jcc4.jar`, c'est-à-dire la version du fichier jar fournie avec la version la plus récente de Db2, ou votre version locale.

- d) A l'étape 2 de l'assistant, vérifiez que l'emplacement du répertoire des fichiers JAR de base de données requis est correctement défini. Cliquez sur **Suivant**.
 - e) Cliquez sur **Terminer** sur la page de récapitulatif pour créer le fournisseur JDBC .
2. Créez des alias d'authentification. Vous créez un alias pour la source de données et un autre pour IBM MQ:
- a) Sélectionnez **Sécurité > Sécurité globale** dans la navigation de la console d'administration WebSphere Application Server traditional 9.0 .
 - b) Sous l'en-tête **Authentification** , développez **Java Authentication and Authorization Service**.
 - c) Cliquez sur **Données d'authentificationJ2C**. La page d'alias d'authentification s'ouvre.
 - d) Créez un alias d'authentification pour votre source de données:
 - i) Cliquez sur **Nouveau**.
 - ii) Entrez les détails de l' **alias**, de l' **ID utilisateur**, du **mot de passe** et de la **description**. Les détails entrés dans les zones **ID utilisateur** et **Mot de passe** doivent correspondre aux détails que vous avez entrés lors de la création de votre utilisateur de base de données. Pour plus d'informations, voir «Configuration de l'accès utilisateur pour le consignateur de base de données Java EE pour MFT», à la page 875.
 - iii) Cliquez sur **OK**.
 - e) Créez un alias d'authentification pour IBM MQ:
 - i) Cliquez sur **Nouveau**.
 - ii) Entrez les détails de l' **alias**, de l' **ID utilisateur**, du **mot de passe** et de la **description**. Les détails entrés dans les zones **ID utilisateur** et **Mot de passe** doivent correspondre à vos paramètres d'utilisateur et de mot de passe pour votre installation IBM MQ .
 - iii) Cliquez sur **OK**.
3. Créez une source de données:
- a) Sélectionnez **Ressources > JDBC > Sources de données** dans la fenêtre de navigation de la console d'administration WebSphere Application Server traditional 9.0 .
 - b) Sélectionnez la liste déroulante **Portée** et remplacez la portée par la valeur appropriée. Par exemple, Node=yourNode , Server=yourServer.
 - c) Créez une source de données à l'aide de l'assistant de la console en cliquant sur **Nouveau**.
 - d) A l'étape 1 de l'assistant, dans la zone **Nom de la source de données** , entrez wmqfte-database et dans la zone **Nom JNDI** , entrez jdbc/wmqfte-database. Cliquez sur **Suivant**.
 - e) A l'étape 2 de l'assistant, utilisez la liste déroulante **Sélectionner un fournisseur JDBC existant** pour sélectionner le fournisseur JDBC créé aux étapes précédentes. Cliquez sur **Suivant**.
 - f) **Db2:** A l'étape 3 de l'assistant, dans la zone **Type de pilote** , entrez 4.
 - g) **Db2:** Entrez les détails dans les zones **Nom de la base de données**, **Nom du serveur** et **Numéro de port** , puis cliquez sur **Suivant**.

Oracle: entrez l'URL de connexion dans la zone **URL** et choisissez l'auxiliaire de magasin de données approprié dans la zone **Nom de classe de l'auxiliaire de magasin de données** .

Oracle RAC: Lors de la connexion à un cluster Oracle Real Application Cluster, l'URL de connexion doit inclure les informations d'hôte nécessaires pour la connexion à toutes les instances disponibles de la base de données.
 - h) A l'étape 4 de l'assistant, sélectionnez le nom de l'alias d'authentification de la source de données que vous avez défini à l'étape 2d dans la liste **Alias d'authentification pour la reprise XA** . Sélectionnez le même nom dans les listes **Alias d'authentification géré par composant** et **Alias d'authentification géré par conteneur** .
 - i) Cliquez sur **Terminer** dans la page récapitulative pour créer la source de données.
4. Facultatif : Vérifiez la configuration de la source de données:

- a) Sélectionnez **Ressources > JDBC > Sources de données** dans la fenêtre de navigation de la console d'administration WebSphere Application Server traditional 9.0 .
 - b) Cliquez sur le bouton **Tester connexion**.
5. Créez une rubrique.
- a) Dans la navigation de la console d'administration WebSphere Application Server traditional 9.0 , cliquez sur **Ressources > JMS > Rubriques**.
 - b) Sélectionnez la liste déroulante **Portée** et remplacez la portée par la valeur appropriée. Par exemple, Node=yourNode, Server=yourServer.
 - c) Cliquez sur **Nouveau**.
 - d) Cliquez sur **IBM MQ**.
 - e) Dans le panneau **Administration** de la page de propriétés de la rubrique, choisissez des valeurs uniques pour les zones **Nom** et **Nom JNDI** , que vous référencerez ultérieurement dans la configuration.
 - f) Dans le panneau **IBM MQ topic** , entrez SYSTEM.FTE/Log/# dans la zone **Nom de la rubrique** .
6. Créez une spécification d'activation:
- a) Dans la navigation de la console d'administration WebSphere Application Server traditional 9.0 , cliquez sur **Ressources > JMS > Spécifications d'activation**.
 - b) Sélectionnez la liste déroulante **Portée** et remplacez la portée par la valeur appropriée. Par exemple, Node=yourNode, Server=yourServer.
 - c) Cliquez sur **Nouveau**.
 - d) Cliquez sur **IBM MQ**.
 - e) A l'étape 1 de l'assistant, choisissez des valeurs uniques pour les zones **Nom** et **Nom JNDI** , que vous référencerez à nouveau ultérieurement dans la configuration.
 - f) A l'étape 1.1, entrez le nom JNDI de la rubrique que vous avez configurée à l'étape 5 dans la zone **Nom JNDI de destination** .
 - g) Dans la liste **Type de destination** , sélectionnez **Rubrique**.
 - h) A l'étape 1.2 de l'assistant, sélectionnez **Abonnement durable**. Entrez SYSTEM.FTE.DATABASELOGGER.AUTO dans la zone **Nom de l'abonnement** .
 - i) A l'étape 2 de l'assistant, sélectionnez **Entrez toutes les informations requises dans cet assistant**.
 - j) A l'étape 2.1, entrez le nom de votre gestionnaire de files d'attente dans la zone **Nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente** .
 - k) A l'étape 2.2, sélectionnez la méthode de transport de votre choix dans la liste **Transport** . Si vous sélectionnez **Liaisons**, aucune autre information n'est requise. Si vous sélectionnez **Client** ou **Liaisons puis client**, entrez les détails pour **Nom d'hôte**, **Port** et **Canal de connexion serveur**.
 - l) Facultatif : Cliquez sur **Tester la connexion** pour vérifier que le gestionnaire de files d'attente est présent. Toutefois, vous pouvez vous attendre à recevoir NOT_AUTHORIZED tant que vous n'avez pas référencé l'alias d'authentification à l'étape 6n.
 - m) Cliquez sur **Sauvegarder**.
 - n) Cliquez sur le nom de la spécification d'activation que vous avez créée. Dans la section **Propriétés générales** de l'onglet **Configuration** , faites défiler l'écran jusqu'au panneau **Avancé** et entrez un nom unique pour identifier votre connexion IBM MQ dans la zone **ID client** . Vous devez effectuer cette étape ou votre connexion est rejetée par IBM MQ avec le code d'erreur JM5CC0101 .
 - o) Si vous avez choisi **Client** comme méthode de transport, accédez au panneau **Paramètres de sécurité** et sélectionnez l'alias d'authentification que vous avez défini à l'étape 8 dans la liste **Alias d'authentification** .
 - p) Cliquez sur **Appliquer**.

- q) Dans la section **Propriétés supplémentaires** de l'onglet **Configuration** , cliquez sur **Propriétés avancées**. Dans la section **Connection Consumer** du panneau **Advanced Properties** , entrez 1 dans la zone **Maximum server sessions** .
- Remarque :** Veillez à effectuer cette étape avant de poursuivre. Si vous ne le faites pas, le consigneur risque de ne pas fonctionner correctement.
- r) Dans la section **Propriétés supplémentaires** de l'onglet **Configuration** , cliquez sur **Propriétés avancées**. Définissez la valeur de **Arrêter le noeud final en cas d'échec de la distribution des messages** sur un minimum de 1.
- Si la valeur de la propriété **_numberOfFailedAttemptsBeforeReject** est supérieure à 1 (voir 9j pour plus d'informations), définissez **Arrêter le noeud final en cas d'échec de la distribution des messages** sur au moins la valeur de la propriété **_numberOfFailedAttemptsBeforeReject** . Cela empêche le noeud final de s'arrêter lorsqu'un message qui ne peut pas être traité (par exemple, un message de journal de transfert syntaxiquement incorrect) est reçu. Pour plus d'informations, voir [Traitement et rejet des erreurs du consigneurMFT](#).
7. Créez une fabrique de connexions de file d'attente.
- Dans la navigation de la console d'administration WebSphere Application Server traditional 9.0 , cliquez sur **Ressources > JMS > Fabriques de connexions de file d'attente**.
 - Sélectionnez la liste déroulante **Portée** et remplacez la portée par la valeur appropriée. Par exemple, Node=yourNode, Server=yourServer.
 - Cliquez sur **Nouveau**.
 - Cliquez sur **IBM MQ**.
 - A l'étape 1 de l'assistant, choisissez des valeurs uniques pour les zones **Nom** et **Nom JNDI** , que vous référencerez à nouveau ultérieurement dans la configuration.
 - A l'étape 2, sélectionnez **Entrez toutes les informations requises dans cet assistant**.
 - A l'étape 2.1, entrez le nom de votre gestionnaire de files d'attente dans la zone **Nom du gestionnaire de files d'attente ou du groupe de partage de files d'attente** .
 - A l'étape 2.2, sélectionnez la méthode de transport de votre choix dans la liste **Transport** . Si vous sélectionnez **Liaisons**, aucune autre information n'est requise. Si vous sélectionnez **Client** ou **Liaisons puis client**, entrez les détails pour **Nom d'hôte**, **Port** et **Canal de connexion serveur**.
 - Facultatif : Cliquez sur **Tester la connexion** pour vérifier que le gestionnaire de files d'attente est présent. Toutefois, vous pouvez vous attendre à recevoir NOT_AUTHORIZED tant que vous n'avez pas référencé l'alias d'authentification à l'étape 7h.
 - Si vous avez sélectionné **Client** ou **Liaisons puis client** comme méthode de transport, cliquez sur le nom de la fabrique de connexions de file d'attente que vous venez de créer. Accédez au panneau **Paramètres de sécurité** de l'onglet **Configuration** et sélectionnez l'alias d'authentification que vous avez défini à l'étape 2e dans les listes **Alias d'authentification pour la reprise XA** et **Alias d'authentification géré par conteneur** .
8. Créez une file d'attente de rejet dans WebSphere Application Server:
- Dans la navigation de la console d'administration WebSphere Application Server traditional 9.0 , cliquez sur **Ressources > JMS > Files d'attente**.
 - Sélectionnez la liste déroulante **Portée** et remplacez la portée par la valeur appropriée. Par exemple, Node=yourNode, Server=yourServer.
 - Cliquez sur **Nouveau**.
 - Cliquez sur **IBM MQ**.
 - Choisissez des valeurs uniques pour les zones **Nom** et **Nom JNDI** , que vous référencerez à nouveau ultérieurement dans la configuration.
 - Entrez SYSTEM.FTE.LOG.RJCT.logger_name dans la zone **Nom de la file d'attente** . Vérifiez que vous avez créé cette file d'attente sur votre gestionnaire de files d'attente de coordination.

- g) Entrez le nom de votre gestionnaire de files d'attente dans la zone **Nom du gestionnaire de files d'attente**.
 - h) Cliquez sur **OK**.
9. Installez l'application de consignateur de base de données JEE:
- a) Dans la console d'administration WebSphere Application Server traditional 9.0, sélectionnez **Applications > Nouvelle application**.
 - b) Sélectionnez la liste déroulante **Portée** et remplacez la portée par la valeur appropriée. Par exemple, Node=yourNode, Server=yourServer.
 - c) Dans la liste des options, sélectionnez **Nouvelle application d'entreprise**.
 - d) Dans la page **Préparation de l'installation de l'application**, sélectionnez le fichier com.ibm.wmqfte.databaselogger.jee.ear ou le fichier com.ibm.wmqfte.databaselogger.jee.oracle.ear dans le répertoire `MQ_INSTALLATION_PATH/mqft/web` de l'installation Managed File Transfer Service, puis cliquez sur **Suivant**.
 - e) Sur l'écran suivant, sélectionnez **Détaillé** pour afficher tous les paramètres et options d'installation, puis cliquez sur **Suivant**.
 - f) Cliquez sur **Suivant** dans les étapes 1 à 4 de l'assistant pour accepter les valeurs par défaut.
 - g) A l'étape 5 de l'assistant, **Liaison des programmes d'écoute pour les beans gérés par message**, accédez à la section **Liaisons d'écoute**. Cliquez sur **Spécification d'activation**.
Entrez les valeurs requises pour les zones suivantes:
 - Nom JNDI de la ressource cible**
Nom JNDI que vous avez spécifié lors de la création d'une spécification d'activation à l'étape 6d.
 - Nom JNDI de destination**
Nom JNDI que vous avez spécifié lors de la création d'une rubrique à l'étape 5d.
 Cliquez sur **Suivant**.
 - h) A l'étape 6 de l'assistant, **Mappage des références de ressource vers les ressources**, entrez les détails dans la zone **Nom JNDI de la ressource cible**. Il s'agit du nom JNDI que vous avez spécifié pour la fabrique de connexions de file d'attente de rejet à l'étape 7c. Cliquez sur **Suivant**.
 - i) A l'étape 7 de l'assistant, **Mappage des références d'entrée d'environnement de ressource vers les ressources**, entrez les détails dans la zone **Nom JNDI de la ressource cible**. Il s'agit du nom JNDI de la file d'attente de rejet que vous avez créée à l'étape 8d. Cliquez sur **Suivant**.
 - j) A l'étape 8 de l'assistant, **Mappage des entrées d'environnement pour les modules EJB**, acceptez la valeur par défaut 1. Cliquez sur **Suivant**.

Oracle RAC: Lors de la connexion à un cluster Oracle Real Application Cluster, vous devez définir la valeur de la propriété `_numberOfFailedAttemptsBeforeReject` sur **au moins 2**. Cette propriété détermine le nombre de fois où le consignateur tente de traiter un message d'audit après un échec. En cas de reprise en ligne de la base de données, au moins un incident est susceptible de se produire. Pour éviter de déplacer inutilement un message dans la file d'attente de rejet, l'augmentation de cette valeur permet d'effectuer une deuxième tentative, ce qui aboutit généralement à la réussite de la connexion à la nouvelle instance de base de données. Si, lors du test, vous constatez que des messages sont toujours déplacés dans la file d'attente de rejet lors de la reprise en ligne de votre instance de base de données, augmentez encore cette valeur: la temporisation du basculement entre les instances peut entraîner plusieurs échecs pour le même message. Toutefois, n'oubliez pas que l'augmentation de cette valeur affecte tous les cas d'échec (par exemple, un message syntaxiquement incorrect) et pas seulement la reprise en ligne de la base de données. Par conséquent, augmentez la valeur avec précaution afin d'éviter les nouvelles tentatives inutiles.
 - k) A l'étape 9 de l'assistant, **Métadonnées pour les modules**, cliquez sur **Suivant**.
 - l) A l'étape 10 de l'assistant, **Récapitulatif**, cliquez sur **Terminer**.

10. Vous pouvez maintenant démarrer l'application à partir de la console d'administration WebSphere Application Server traditional 9.0 :
 - a) Sélectionnez **Applications > Types d'application > WebSphere** dans la navigation de la console.
 - b) Cochez la case de l'application d'entreprise **Logger** dans la table des collections et cliquez sur **Démarrer**.

Configuration de l'accès utilisateur pour le consignateur de base de données Java EE pour MFT

Lorsque vous configurez le consignateur de base de données Java Platform, Enterprise Edition (Java EE) pour Managed File Transfer, vous avez besoin de comptes utilisateur pour accéder à IBM MQ, à votre base de données et à votre système d'exploitation. Le nombre d'utilisateurs du système d'exploitation requis dépend du nombre de systèmes que vous utilisez pour héberger ces composants.

Pourquoi et quand exécuter cette tâche

Le nombre et le type de comptes utilisateur dont vous avez besoin pour exécuter le consignateur de base de données Java EE dépendent du nombre de systèmes que vous utilisez. Les comptes utilisateur sont requis pour accéder aux trois environnements suivants:

- Système d'exploitation local
- IBM MQ
- Base de données

Vous pouvez installer le consignateur de base de données JEE, IBM MQ et votre base de données sur un seul système ou sur plusieurs systèmes. Les composants peuvent être installés dans les exemples de topologies suivants:

Le consignateur de base de données Java EE , IBM MQ et la base de données se trouvent tous sur le même système

Vous pouvez définir un utilisateur de système d'exploitation unique à utiliser avec les trois composants. Le consignateur utilise le mode Liaisons pour se connecter à IBM MQ et une connexion native pour se connecter à la base de données.

Consignateur de base de données Java EE et IBM MQ sur un système, la base de données sur un système distinct

Vous créez deux utilisateurs pour cette configuration: un utilisateur du système d'exploitation sur le système exécutant le consignateur et un utilisateur du système d'exploitation disposant d'un accès distant à la base de données sur le serveur de base de données. Le consignateur utilise le mode Liaisons pour se connecter à IBM MQ et une connexion client pour accéder à la base de données.

Le consignateur de base de données Java EE sur un système, IBM MQ sur un autre système, la base de données sur un autre système

Vous créez trois utilisateurs pour cette configuration: un utilisateur du système d'exploitation pour démarrer le serveur d'applications, un utilisateur IBM MQ pour accéder aux files d'attente et aux rubriques utilisées et un utilisateur du serveur de base de données pour accéder aux tables de base de données et les insérer dans ces dernières. Le consignateur utilise le mode client pour accéder à IBM MQ et une connexion client pour accéder à la base de données.

Par exemple, le reste de ces instructions suppose que l'utilisateur est appelé `fte1og`, mais vous pouvez utiliser n'importe quel nom d'utilisateur, nouveau ou existant. Configurez les droits utilisateur comme suit:

Procédure

1. Assurez-vous que l'utilisateur du système d'exploitation possède son propre groupe et qu'il ne fait partie d'aucun groupe disposant de droits étendus sur le gestionnaire de files d'attente de coordination. L'utilisateur ne doit pas faire partie du groupe `mqm`. Sur certaines plateformes, le groupe de personnel reçoit également automatiquement l'accès au gestionnaire de files d'attente ; l'utilisateur du consignateur ne doit pas se trouver dans le groupe de personnel. Vous pouvez afficher les enregistrements de droits d'accès pour le gestionnaire de files d'attente lui-même et pour les

objets qu'il contient à l'aide de la IBM MQ Explorer. Cliquez avec le bouton droit de la souris sur l'objet et sélectionnez **Droits sur les objets > Gérer les enregistrements de droits d'accès**. Sur la ligne de commande, vous pouvez utiliser les commandes `dspmqaut` (`display authority`) ou `dmpmqaut` (`dump authority`).

2. Utilisez la fenêtre **Gérer les enregistrements de droits d'accès** dans le IBM MQ Explorer ou la commande `setmqaut` (`grant or revoke authority`) pour ajouter des droits d'accès au groupe de l'utilisateur IBM MQ (sous AIX, les droits d'accès IBM MQ sont associés à des groupes uniquement et non à des utilisateurs individuels). Les autorités requises sont les suivantes:

- CONNECT et INQUIRE sur le gestionnaire de files d'attente (les bibliothèques IBM MQ Java requièrent l'autorisation INQUIRE pour fonctionner).
- Droit SUBSCRIBE sur SYSTEM.FTE .
- Droit PUT sur le système SYSTEM.FTE.LOG.RJCT.*nom_journal* file d'attente.

Les noms de file d'attente de rejet et de commandes indiqués sont les noms par défaut. Si vous avez choisi des noms de file d'attente différents lors de la configuration des files d'attente du consignateur, ajoutez les droits à ces noms de file d'attente à la place.

3. Effectuez la configuration utilisateur de base de données spécifique à la base de données que vous utilisez.

- Si votre base de données est Db2, procédez comme suit:

Remarque : Il existe plusieurs mécanismes de gestion des utilisateurs de base de données avec Db2. Ces instructions s'appliquent au schéma par défaut basé sur les utilisateurs du système d'exploitation.

- Vérifiez que l'utilisateur `fte1log` ne fait partie d'aucun groupe d'administration Db2 (par exemple, `db2iadm1`, `db2fadm1` ou `dasadm1`).
- Accordez à l'utilisateur le droit de se connecter à la base de données et le droit de sélectionner, d'insérer et de mettre à jour les tables que vous avez créées dans le cadre de l' étape 2: création des tables de base de données requises.

- Si votre base de données est Oracle, procédez comme suit:

- Vérifiez que l'utilisateur `fte1log` ne fait partie d'aucun groupe d'administration Oracle (par exemple, `ora_dba` sous Windows ou `dba` sous AIX and Linux).
- Accordez à l'utilisateur le droit de se connecter à la base de données et le droit de sélectionner, d'insérer et de mettre à jour les tables que vous avez créées dans le cadre de l' étape 2: création des tables de base de données requises.

Migration du consignateur de base de données autonome vers le consignateur de base de données Java EE pour MFT


Vous pouvez effectuer une migration depuis le consignateur de base de données autonome vers le consignateur de base de données Java EE . Vous devez arrêter le consignateur de base de données autonome et installer le consignateur de base de données JEE. Pour éviter de perdre ou de dupliquer des entrées de journal, vous devez arrêter la publication de messages dans SYSTEM.FTE avant d'arrêter le consignateur de base de données autonome et de le redémarrer après avoir installé le consignateur de base de données Java EE . Sauvegardez votre base de données avant la migration.

Pourquoi et quand exécuter cette tâche

Procédure

1. Avant d'arrêter la base de données, exécutez la commande MQSC suivante sur votre gestionnaire de files d'attente de coordination: `ALTER QM PSMODE (COMPAT)`

Les messages ne sont plus publiés dans SYSTEM.FTE/Log . Attendez que le consignateur ait traité tous les messages de son abonnement. Par défaut, cet abonnement est appelé SYSTEM.FTE.LOGGER.AUTO.

2. Arrêtez le consignateur de base de données à l'aide de la commande **fteStopLogger** .
3. Sauvegardez la base de données à l'aide des outils fournis avec le logiciel de base de données.
4. Supprimez l'abonnement appartenant au consignateur de base de données autonome.
Par défaut, cet abonnement est appelé SYSTEM.FTE.LOGGER.AUTO.
5. Si votre schéma de base de données est à une version antérieure, vous devez le migrer à chaque niveau suivant dans l'ordre. Par exemple, si votre schéma de base de données est au niveau V7.0.1 et que vous migrez vers V7.0.4, vous devez migrer votre schéma depuis V7.0.1 vers V7.0.2, puis depuis V7.0.2 vers V7.0.3, puis depuis V7.0.3 vers V7.0.4. Migrez votre schéma de base de données de la version *old* vers la version *new*, où *old* et *new* sont des variables qui décrivent une version de schéma, en effectuant l'une des actions suivantes pour chaque version du schéma que vous devez migrer:
 -  Si votre base de données est Db2 sur z/OS et que vous migrez entre les schémas V7.0.2 et V7.0.3 ou entre les schémas V7.0.3 et V7.0.4 , vous devez créer un nouveau schéma de base de données et y copier vos données existantes. Pour plus d'informations, reportez-vous à la documentation Db2.
 - Si votre base de données n'est pas Db2 ou si vous avez créé votre base de données avec une taille de page supérieure à 8K, vous pouvez migrer le schéma de la même manière que pour les autres versions, en procédant comme suit.
 - Si vous effectuez une migration entre des tables de base de données dans d'autres cas, procédez comme suit:
 - a. Choisissez le fichier correspondant à votre plateforme de base de données et dont le nom inclut la chaîne *old-new*. Ce fichier se trouve dans le répertoire `MQ_INSTALLATION_PATH/mqft/sql` de l'installation des outils distants et de la documentation.
 - b. Si vous avez apporté des modifications au schéma initial, examinez le fichier de migration pour vous assurer qu'il sera compatible avec votre base de données modifiée.
 - c. Exécutez le fichier SQL sur votre base de données.
6. Installez le fichier EAR du consignateur de base de données Java EE .
7. Déployez le consignateur de base de données Java EE . Pour plus d'informations, voir [«Installation du consignateur de base de données Java EE pour MFT»](#), à la page 866.
8. Exécutez la commande MQSC suivante sur votre gestionnaire de files d'attente de coordination: ALTER QMGR PSMODE(ENABLED)
Cela permet de publier des messages dans SYSTEM.FTE/Log .

Résultats

Configuration du pont Connect:Direct

Configurez le pont Connect:Direct pour le transfert de fichiers entre un réseau Managed File Transfer et un réseau Connect:Direct . Les composants du pont Connect:Direct sont un noeud Connect:Direct et un agent Managed File Transfer dédié à la communication avec ce noeud. Cet agent est appelé agent de pont Connect:Direct .

Avant de commencer

L'agent et le noeud qui composent le pont Connect:Direct doivent se trouver sur le même système ou avoir accès au même système de fichiers, par exemple via un montage NFS partagé. Ce système de fichiers est utilisé pour stocker temporairement les fichiers lors des transferts de fichiers impliquant le pont Connect:Direct, dans un répertoire défini par le paramètre **cdTmpDir**. L'agent de pont Connect:Direct et le noeud de pont Connect:Direct doivent pouvoir traiter ce répertoire en utilisant le même chemin d'accès. Par exemple, si l'agent et le noeud se trouvent sur des systèmes Windows distincts, les systèmes doivent utiliser la même lettre d'unité pour monter le système de fichiers partagé. Les configurations suivantes permettent à l'agent et au noeud d'utiliser le même chemin d'accès :

- L'agent et le noeud se trouvent sur le même système, qui exécute Windows ou Linux for x86-64
- L'agent se trouve sous Linux for x86-64 et le noeud sous AIX
- L'agent se trouve sur un système Windows et le noeud sur un autre système Windows

Les configurations suivantes ne permettent pas à l'agent et au noeud d'utiliser le même chemin d'accès :

- L'agent se trouve sous Linux for x86-64 et le noeud sous Windows
- L'agent se trouve sous Windows et le noeud sous UNIX

Tenez compte de cette restriction lorsque vous planifiez votre installation du pont Connect:Direct.

Pour plus de détails sur les versions de système d'exploitation prises en charge pour le pont Connect:Direct , voir la page Web [System Requirements for IBM MQ](#).

Pourquoi et quand exécuter cette tâche

Un agent de pont Connect:Direct est un agent Managed File Transfer dédié à la communication avec un noeud Connect:Direct .

Par défaut, l'agent de pont Connect:Direct utilise le protocole TCP/IP pour se connecter au noeud Connect:Direct . Si vous souhaitez établir une connexion sécurisée entre votre agent de pont Connect:Direct et le noeud Connect:Direct , vous pouvez utiliser le protocole SSL ou TLS.

Procédure

1. Choisissez les systèmes d'exploitation pour l'agent de pont et le noeud Connect:Direct :
 - a) Choisissez un système exécutant Windows ou Linux sur x86-64 pour installer l'agent de pont Connect:Direct .
 - b) Choisissez un système d'exploitation pris en charge par Connect:Direct for Windows ou Connect:Direct for UNIX pour installer le noeud de pont Connect:Direct .

2. Choisissez et configurez un noeud Connect:Direct .

Vous devez avoir installé un noeud Connect:Direct avant de suivre ces instructions.

- a) Choisissez un noeud Connect:Direct avec lequel l'agent Managed File Transfer doit communiquer.
- b) Vérifiez la mappe de réseau pour le noeud Connect:Direct que vous avez choisi. Si la mappe de réseau contient des entrées pour les noeuds distants s'exécutant sur un système d'exploitation Windows , vous devez vous assurer que ces entrées indiquent que les noeuds s'exécutent sur Windows.



Si le noeud Connect:Direct que vous avez sélectionné pour le pont Connect:Direct est en cours d'exécution sur Windows, utilisez le demandeur Connect:Direct pour éditer la mappe de réseau. Vérifiez que la zone **Système d'exploitation** pour tous les noeuds distants qui s'exécutent sur Windows est définie sur **Windows**.

3. Créez et configurez un agent de pont Connect:Direct .

- a) Créez un agent de pont Connect:Direct à l'aide de la commande **fteCreateCDAgent** .
 - Vous devez indiquer une valeur pour le paramètre **cdNode** . Ce paramètre indique le nom utilisé par l'agent pour le noeud Connect:Direct qui fait partie du pont Connect:Direct . Utilisez le nom du noeud Connect:Direct que vous avez choisi dans la section précédente.
 - Indiquez des valeurs pour les paramètres **cdNodeHost** et **cdNodePort** , qui définissent le noeud Connect:Direct avec lequel l'agent communique.

Si vous n'indiquez pas de valeur pour le paramètre **cdNodeHost** , le nom d'hôte ou l'adresse IP du système local est utilisé. Si vous n'indiquez pas de valeur pour le paramètre **cdNodePort** , la valeur 1363 est utilisée.

- Utilisez éventuellement les informations de la rubrique [fteCreateAgent](#) pour déterminer si vous devez spécifier une valeur pour le paramètre **cdTmpDir** .


- b) Mappez les données d'identification de l'utilisateur utilisées par Managed File Transfer aux données d'identification de l'utilisateur sur un noeud Connect:Direct . Vous pouvez mapper des données d'identification à l'aide de l'une des méthodes suivantes:
- Créez un fichier `ConnectDirectCredentials.xml` pour définir les informations de mappage des données d'identification. Pour plus d'informations, voir «[Mappage des données d'identification pour Connect:Direct à l'aide du fichier ConnectDirectCredentials.xml](#)», à la page 880.
 - Ecrivez un exit utilisateur pour effectuer un mappage de données d'identification pour votre pont Connect:Direct . Pour plus d'informations, voir «[Mappage des données d'identification pour Connect:Direct à l'aide de classes d'exit](#)», à la page 882.
4. Configurez le fichier `ConnectDirectNodeProperties.xml` pour inclure des informations sur les noeuds Connect:Direct distants.

Vous devez avoir créé un agent de pont Connect:Direct avant de suivre ces instructions.

Éditez le modèle `ConnectDirectNodeProperties.xml` dans le répertoire de configuration de l'agent de pont Connect:Direct . Pour chaque noeud ou groupe de noeuds Connect:Direct sur lequel vous souhaitez définir des informations, procédez comme suit:

- a) Dans l'élément `nodeProperties` , créez un élément `node` .
- b) Ajoutez un attribut `name` à l'élément `node` . Spécifiez la valeur de cet attribut en tant que modèle correspondant au nom d'un ou de plusieurs noeuds Connect:Direct distants.
- c) Facultatif : Ajoutez un attribut `pattern` à l'élément `node` qui spécifie le type de canevas que la valeur de l'attribut `name` est. Les valeurs admises sont `regex` et `wildcard`. L'option par défaut est `wildcard`.
- d) Ajoutez un attribut `type` à l'élément `node` qui spécifie le système d'exploitation sur lequel s'exécutent les noeuds Connect:Direct distants spécifiés par l'attribut `name` .

Les valeurs suivantes sont valides :

- Windows -le noeud s'exécute sur Windows
- UNIX -le noeud s'exécute sur AIX and Linux
-  z/OS, zos, os/390 ou os390 -le noeud s'exécute sur z/OS

La valeur de cet attribut n'est pas sensible à la casse. Les transferts sur des noeuds distants sur d'autres systèmes d'exploitation ne sont pas pris en charge par le pont Connect:Direct.

Pour plus d'informations, voir [Connect:Direct node properties file format](#).

5. Configurez une connexion sécurisée entre l'agent de pont Connect:Direct et le noeud Connect:Direct . Pour obtenir un exemple de cette procédure, voir [Configuration de SSL ou TLS entre l'agent de pont Connect:Direct et le noeud Connect:Direct](#).

Tâches associées

[Traitement des incidents liés au pont Connect:Direct](#)

[Configuration de SSL ou TLS entre l'agent de pont Connect:Direct et le noeud Connect:Direct](#)

[Transfert d'un fichier vers un noeud Connect:Direct](#)

[Transfert d'un fichier à partir d'un noeud Connect:Direct](#)

 [Transfert de plusieurs fichiers à partir d'un noeud Connect:Direct](#)

Référence associée

[Le pont Connect:Direct](#)

Mappage des données d'identification pour Connect:Direct

Mappez les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur un noeud Connect:Direct à l'aide de la fonction de mappage des données d'identification par défaut de l'agent de pont Connect:Direct ou en écrivant votre propre exit utilisateur. Managed File Transfer fournit un exemple d'exit utilisateur qui effectue le mappage des données d'identification de l'utilisateur.

Tâches associées

«Mappage des données d'identification pour Connect:Direct à l'aide du fichier `ConnectDirectCredentials.xml`», à la page 880

Mappez les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur les noeuds Connect:Direct à l'aide de la fonction de mappage des données d'identification par défaut de l'agent de pont Connect:Direct . Managed File Transfer fournit un fichier XML que vous pouvez éditer pour inclure vos données d'identification.

«Mappage des données d'identification pour Connect:Direct à l'aide de classes d'exit», à la page 882
Si vous ne souhaitez pas utiliser la fonction de mappage des données d'identification par défaut de l'agent de pont Connect:Direct , vous pouvez mapper les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur un noeud Connect:Direct en écrivant votre propre exit utilisateur. La configuration de vos propres exits utilisateur de mappage de données d'identification désactive la fonction de mappage de données d'identification par défaut.

Référence associée

[Interface CDCredentialExit.java](#)

[Format de fichier de données d'identification Connect:Direct](#)

Mappage des données d'identification pour Connect:Direct à l'aide du fichier `ConnectDirectCredentials.xml`

Mappez les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur les noeuds Connect:Direct à l'aide de la fonction de mappage des données d'identification par défaut de l'agent de pont Connect:Direct . Managed File Transfer fournit un fichier XML que vous pouvez éditer pour inclure vos données d'identification.

Pourquoi et quand exécuter cette tâche

Une fois qu'un agent de pont Connect:Direct a été créé à l'aide de la commande **fteCreateCDAgent** , un fichier `ConnectDirectCredentials.xml` doit être créé manuellement. Avant de pouvoir utiliser un agent de pont Connect:Direct , vous devez éditer ce fichier pour inclure les informations d'hôte, d'utilisateur et de données d'identification. Pour plus d'informations, voir [Format de fichier de données d'identification Connect:Direct](#). Par défaut, ce fichier est chargé à partir du répertoire de base de l'utilisateur en cours, `/home/fteuser/ConnectDirectCredentials.xml` par exemple. Pour utiliser un autre emplacement, spécifiez-le à l'aide de l'élément `<credentialsFile>` dans le fichier `ConnectDirectNodeProperties.xml` .

Procédure

1. Vérifiez que l'attribut `name` dans l'élément `<tns:pnode name="Connect:Direct node host" pattern="wildcard">` contient la valeur du nom du noeud Connect:Direct auquel l'agent de pont Connect:Direct se connecte. Cette valeur doit être identique à celle que vous spécifiez pour le paramètre **fteCreateCDAgent -cdNode** .

La valeur de l'attribut `pattern` peut être `wildcard` ou `regex`. Si cet attribut n'est pas spécifié, la valeur par défaut est `wildcard`.

2. Insérez l'ID utilisateur et les données d'identification dans le fichier en tant qu'éléments enfant de `<tns:pnode>`.

Vous pouvez insérer une ou plusieurs instances de l'élément `<tns:user>` suivant dans le fichier:

```
<tns:user name="name"
          pattern="pattern"
          ignorecase="ignorecase"
          cdUserId="cdUserId"
          cdPassword="cdPassword"
          pnodeUserId="pnodeUserId"
          pnodePassword="pnodePassword">
</tns:user>
```

où :

- *name* est un modèle qui correspond à l'ID utilisateur MQMD associé à la demande de transfert MFT .
- *pattern* indique si le modèle spécifié pour l'attribut *name* est une expression générique ou une expression régulière Java . La valeur de l'attribut *pattern* peut être wildcard ou regex. Si cet attribut n'est pas spécifié, la valeur par défaut est wildcard.
- *ignorecase* indique si le modèle spécifié par l'attribut *name* doit être traité comme sensible à la casse. Si cet attribut n'est pas spécifié, la valeur par défaut est true.
- *cdUserId* est l'ID utilisateur utilisé par l'agent de pont Connect:Direct pour se connecter au noeud Connect:Direct spécifié par l'attribut *name* de l'élément <tns : pnode> . Si possible, vérifiez que *cdUserId* est un ID administrateur Connect:Direct . Si *cdUserId* ne peut pas être un administrateur Connect:Direct , vérifiez que l'ID utilisateur dispose des droits fonctionnels suivants sur le noeud de pont Connect:Direct :
 - Pour un noeud Windows , définissez les droits suivants. Cet exemple est formaté avec des retours chariot pour faciliter la lisibilité:

```
View Processes in the TCQ      value: yes
Issue the copy receive, copy send, run job, and run task Process statements
Issue the submit Process statement      value: yes
Monitor, submit, change, and delete all Processes      value: all
Access Process statistics      value: all
Use the trace tool or issue traceon and traceoff commands      value: yes
Override Process options such as file attributes and remote node ID      value: yes
```

- Pour un noeud AIX ou Linux , définissez les paramètres suivants dans le fichier *userfile.cfg* :

```
pstmt.copy      value: y
pstmt.upload    value: y
pstmt.download  value: y
pstmt.runjob    value: y
pstmt.runtask   value: y
cmd.submit      value: y
pstmt.submit    value: y
cmd.chgproc     value: y
cmd.delproc     value: y
cmd.flsproc     value: y
cmd.selproc     value: a
cmd.selstats    value: a
cmd.trace       value: y
snode.ovrd      value: y
```

- *cdPassword* est le mot de passe associé à l'ID utilisateur spécifié par l'attribut *cdUserId* .
- Vous pouvez éventuellement spécifier l'attribut *pnodeUserId* . La valeur de cet attribut est l'ID utilisateur utilisé par le noeud Connect:Direct spécifié par l'attribut *name* de l'élément <tns : pnode> pour soumettre le processus Connect:Direct . Si vous ne spécifiez pas l'attribut *pnodeUserId* , le noeud Connect:Direct utilise l'ID utilisateur spécifié par l'attribut *cdUserId* pour soumettre le processus Connect:Direct .
- Vous pouvez éventuellement spécifier l'attribut *pnodePassword*. La valeur de cet attribut est le mot de passe associé à l'ID utilisateur spécifié par l'attribut *pnodeUserId* .

Si aucun élément utilisateur ne correspond à l'ID utilisateur MQMD, le transfert échoue.

3. Facultatif : Vous pouvez inclure un ou plusieurs éléments <tns : snode> en tant qu'éléments enfant de l'élément <tns : user> . L'élément <tns : snode> spécifie les données d'identification utilisées par

le noeud Connect:Direct qui fait partie du pont Connect:Direct . Ces données d'identification sont l'ID utilisateur et le mot de passe utilisés par le noeud de pont Connect:Direct pour se connecter au noeud Connect:Direct qui est la source ou la destination du transfert de fichiers.

Insérez un ou plusieurs des éléments suivants dans le fichier:

```
<tns:snode name="name"
           pattern="pattern"
           userId="userId"
           password="password" />
```

où :

- *name* est un modèle qui correspond au nom du noeud Connect:Direct qui est la source ou la destination du transfert de fichiers.
- *pattern* indique si le modèle spécifié pour l'attribut name est une expression générique ou une expression régulière Java . La valeur de l'attribut de canevas peut être wildcard ou regex. Si cet attribut n'est pas spécifié, la valeur par défaut est wildcard.
- *userId* est l'ID utilisateur utilisé par le noeud Connect:Direct spécifié par l'attribut name de l'élément <tns:pnode> pour se connecter à un noeud Connect:Direct qui correspond au modèle spécifié par l'attribut name de <tns:snode>.
- *password* est le mot de passe associé à l'ID utilisateur spécifié par l'attribut userId .

Si aucun élément <tns:snode> ne correspond au noeud secondaire du transfert de fichiers, le transfert n'échoue pas. Le transfert est démarré et aucun ID utilisateur ni mot de passe n'est spécifié pour être utilisé avec le noeud secondaire.

Résultats

Lors de la recherche d'une correspondance de modèle pour les noms d'utilisateur ou les noms de noeud Connect:Direct , l'agent de pont Connect:Direct effectue des recherches depuis le début du fichier jusqu'à la fin du fichier. La première correspondance trouvée est celle qui est utilisée.

Tâches associées

«Configuration du pont Connect:Direct», à la page 877

Configurez le pont Connect:Direct pour le transfert de fichiers entre un réseau Managed File Transfer et un réseau Connect:Direct . Les composants du pont Connect:Direct sont un noeud Connect:Direct et un agent Managed File Transfer dédié à la communication avec ce noeud. Cet agent est appelé agent de pont Connect:Direct .

Référence associée

[Format de fichier de données d'identification Connect:Direct](#)

[fteCreateCDAgent: créez un agent de pont Connect:Direct](#)

Mappage des données d'identification pour Connect:Direct à l'aide de classes d'exit

Si vous ne souhaitez pas utiliser la fonction de mappage des données d'identification par défaut de l'agent de pont Connect:Direct , vous pouvez mapper les données d'identification de l'utilisateur dans Managed File Transfer aux données d'identification de l'utilisateur sur un noeud Connect:Direct en écrivant votre propre exit utilisateur. La configuration de vos propres exits utilisateur de mappage de données d'identification désactive la fonction de mappage de données d'identification par défaut.

Pourquoi et quand exécuter cette tâche

Les exits utilisateur que vous créez pour le mappage des données d'identification Connect:Direct doivent implémenter l'interface `com.ibm.wmqfte.exitroutine.api.ConnectDirectCredentialExit`. Pour plus d'informations, voir [CDCredentialExit.java](#) interface.

Configuration de IBM MQ Console et de REST API

Le serveur mqweb qui héberge IBM MQ Console et REST API est fourni avec une configuration par défaut. Pour utiliser l'un ou l'autre de ces composants, un certain nombre de tâches de configuration doivent être effectuées, telles que la configuration de la sécurité pour permettre aux utilisateurs de se connecter. Cette rubrique décrit toutes les options de configuration disponibles.

Procédure

- [«Configuration de base pour le serveur mqweb»](#), à la page 883
- [«Configuration de la sécurité»](#), à la page 889
- [«Configuration du nom d'hôte HTTP»](#), à la page 889
- [«Configuration des ports HTTP et HTTPS»](#), à la page 890
- [«Configuration du délai d'attente de réponse»](#), à la page 891
- [«Configuration du démarrage automatique»](#), à la page 892
- [«Configuration de la journalisation»](#), à la page 893
- [«Configuration du jeton LTPA»](#), à la page 897
- [«Configuration du comportement de connexion du gestionnaire de files d'attente éloignées pour IBM MQ Console»](#), à la page 899
- [«Configuration de la passerelle administrative REST API»](#), à la page 901
- [«Configuration de messaging REST API»](#), à la page 902
- [«Configuration de REST API pour MFT»](#), à la page 909
- [«Optimisation de la machine virtuelle Java du serveur mqweb»](#), à la page 914
- [«Structure de fichier du composant d'installation IBM MQ Console et REST API»](#), à la page 916





Configuration de base pour le serveur mqweb

Avant de commencer à utiliser REST API ou IBM MQ Console, vous devez installer les composants appropriés et configurer le serveur mqweb qui héberge REST API ou IBM MQ Console.

Pourquoi et quand exécuter cette tâche

La procédure de cette tâche se concentre sur une configuration de base pour le serveur mqweb afin que vous puissiez démarrer rapidement avec REST API et IBM MQ Console. 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 serveur mqweb, voir [Sécurité IBM MQ Console et REST API](#).

Remarque : Vous devez avoir accès au fichier `mqwebuser.xml` pour exécuter cette procédure:

-  Sous z/OS, vous devez être un utilisateur disposant d'un accès en écriture au fichier `mqwebuser.xml`.
-  Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#) pour accéder au fichier `mqwebuser.xml`.
-   Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server.

Procédure

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

- **AIX** Sous AIX, installez l'ensemble de fichiers `mqm.web.rte`. Pour plus d'informations sur l'installation d'ensembles de fichiers sous AIX, voir [Tâches d'installation d'AIX](#).
 - **IBM i** Sous IBM i, installez le composant WEB. Pour utiliser cette fonction, vous devez également installer les prérequis 5724L26 IBM MQ Java Messaging and Web Services et 5770JV1 Java SE 8. Pour plus d'informations sur l'installation de fonctions sous IBM i, voir [Tâches d'installation d'IBM i](#).
 - **Linux** Sous Linux, installez le composant MQSeriesWeb. Pour plus d'informations sur l'installation de composants sous Linux, voir [Tâches d'installation d'Linux](#).
 - **V 9.4.0** Depuis IBM MQ 9.4.0, vous pouvez également exécuter le serveur mqweb dans une installation IBM MQ Web Server autonome sous Linux. Pour plus d'informations sur l'installation du IBM MQ Web Server, voir [Installation du IBM MQ Web Server autonome](#).
 - **Windows** Sous Windows, installez la fonction Web Administration. Pour plus d'informations sur l'installation de fonctions sous Windows, voir [Tâches d'installation d'Windows](#).
 - **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. Créez le serveur mqweb qui héberge IBM MQ Console et REST API.
- **z/OS** Sous z/OS, exécutez le script `crtmqweb`.
Ce script crée le répertoire utilisateur WebSphere Liberty qui contient la configuration du serveur mqweb et les fichiers journaux. Pour plus d'informations sur l'exécution du script `crtmqweb`, voir «Création du serveur mqweb», à la page 998.
 - **V 9.4.0** **Linux** Dans une installation IBM MQ Web Server autonome, suivez les étapes de la rubrique «Configuration du IBM MQ Web Server autonome», à la page 887.
 - Dans tous les autres environnements, vous n'avez pas besoin d'effectuer d'actions pour créer le serveur mqweb.
3. **z/OS**
Sous z/OS, créez une procédure cataloguée pour démarrer le serveur mqweb.
Pour plus d'informations, voir «Creating a procedure for the mqweb server», à la page 1000.
4. Remplacez le fichier de configuration existant, `mqwebuser.xml`, par l'exemple de fichier de registre de base configuré pour offrir la sécurité de base. Copiez le fichier `basic_registry.xml` du répertoire `MQ_INSTALLATION_PATH/web/mq/samp/configuration` vers le répertoire approprié pour votre système et renommez le fichier en `mqwebuser.xml`:
- Dans une installation IBM MQ, copiez le fichier dans le répertoire suivant:
 - **Linux** **AIX** Sous AIX and Linux : `/var/mqm/web/installations/installationName/servers/mqweb`
 - **Windows** Sous Windows :
`MQ_DATA_PATH\web\installations\installationName\servers\mqweb`
Où `MQ_DATA_PATH` est le chemin de données IBM MQ, ce chemin est le chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.
 - **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` pour créer la définition de serveur mqweb.

- V 9.4.0
Linux
 Dans une installation IBM MQ Web Server autonome:
`MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST/servers/mqweb`
 où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel la variable d'environnement **MQ_OVERRIDE_DATA_PATH** pointe.

L'exemple de fichier `basic_registry.xml` configure quatre utilisateurs:

MQADMIN

Administrateur membre du rôle MQWebAdmin .

mqreader

Administrateur en lecture seule qui est membre du rôle MQWebAdminRO .

mftadmin

Administrateur membre du rôle MFTWebAdmin .

Mftreader

Administrateur en lecture seule qui est membre du rôle MFTWebAdminRO .

Tous les utilisateurs sont également membres du rôle MQWebUser .

Pour plus d'informations sur les rôles disponibles, voir [Rôles sur IBM MQ Console et REST API](#)

- Facultatif : Editez le fichier `mqwebuser.xml` pour ajouter d'autres utilisateurs et groupes. Affectez à ces utilisateurs et à ces groupes les rôles appropriés pour être autorisés à utiliser REST API ou IBM MQ Console. 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](#).

Remarque :

- z/OS
 Sous z/OS, si vous ajoutez des utilisateurs au rôle MQWebUser , vous devez également accorder à l'ID utilisateur de la tâche démarrée mqweb un autre accès utilisateur aux ID utilisateur ayant le rôle MQWebUser . Exemple :

```
RDEFINE MQADMIN h1q.ALTERNATE.USER.userId UACC(NONE)
PERMIT h1q.ALTERNATE.USER.userId CLASS(MQADMIN) ACCESS(UPDATE) ID(mqwebUserId)
```

- Multi
z/OS
 Pour commencer à utiliser le messaging REST API, vous devez ajouter un utilisateur au fichier `mqwebuser.xml` . Cet utilisateur doit avoir le même nom qu'un utilisateur IBM MQ existant sur votre système. En suivant le même format que les autres utilisateurs dans le fichier XML, ajoutez l'ID utilisateur et un mot de passe après la ligne suivante dans le fichier XML:
`<user name="mftreader" password="mftreader"/>`
- Définissez votre environnement pour qu'il pointe vers la configuration du serveur mqweb.

- z/OS
 Sous z/OS, définissez la variable d'environnement WLP_USER_DIR de sorte qu'elle pointe vers votre configuration de serveur mqweb, en entrant la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

Où `WLP_user_directory` est le nom du répertoire transmis à la commande **crtmqweb** .

Exemple :

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

Pour plus d'informations, voir «Création du serveur mqweb», à la page 998.

- V 9.4.0
Linux
 Dans une installation IBM MQ Web Server autonome, définissez la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Par exemple, si vous avez choisi d'utiliser /var/mqweb comme répertoire de données IBM MQ Web Server , exécutez la commande suivante:

```
export MQ_OVERRIDE_DATA_PATH=/var/mqweb
```

- Dans tous les autres environnements, vous n'avez pas besoin d'effectuer d'actions pour définir votre environnement.
7. Par défaut, REST API et IBM MQ Console sont disponibles uniquement à partir du même hôte que le serveur mqweb. Activez les connexions distantes au serveur mqweb en entrant la commande suivante:

```
setmqweb properties -k httpHost -v hostname
```

Où *hostname* indique l'adresse IP, le nom d'hôte du serveur de noms de domaine (DNS) avec le suffixe de nom de domaine ou le nom d'hôte DNS du serveur sur lequel IBM MQ est installé. Utilisez un astérisque, *, entre guillemets, pour spécifier toutes les interfaces réseau disponibles, comme illustré dans l'exemple suivant:

```
setmqweb properties -k httpHost -v "*" 
```

8. Facultatif : Par défaut, administrative REST API for MFT n'est pas activé. Si vous souhaitez utiliser cette fonction, vous devez l'activer et configurer un gestionnaire de files d'attente de coordination:

- a) Activez administrative REST API for MFT en entrant la commande suivante:

```
setmqweb properties -k mqRestMftEnabled -v true
```

- b) Configurez le gestionnaire de files d'attente qui est le gestionnaire de files d'attente de coordination en entrant la commande suivante:

```
setmqweb properties -k mqRestMftCoordinationQmgr -v qmgrName
```


Où *qmgrName* est le nom du gestionnaire de file d'attente de coordination.

- c) Pour activer les appels POST, configurez le gestionnaire de files d'attente qui est le gestionnaire de files d'attente de commandes en entrant la commande suivante:


```
setmqweb properties -k mqRestMftCommandQmgr -v qmgrName
```

Où *qmgrName* est le nom du gestionnaire de files d'attente de commandes.


9. Démarrez le serveur mqweb qui prend en charge REST API et IBM MQ Console:

-  Sous AIX, Linux, and Windows, en tant qu' utilisateur privilégié, entrez la commande suivante:

```
strmqweb
```

-  Sous IBM i, en tant qu' utilisateur privilégié, entrez la commande suivante dans Qshell:

```
/QIBM/ProdData/mqm/bin/strmqweb
```

-  Sous z/OS, démarrez la procédure que vous avez créée dans «[Creating a procedure for the mqweb server](#)», à la page 1000.

Les messages suivants sont envoyés à la définition de données STDOUT pour indiquer que le démarrage du serveur mqweb a abouti.

```
[AUDIT ] MQWB2019I: MQ Console level: 9.2.4 - V924-CD924-L211028
[AUDIT ] MQWB0023I: MQ REST API level: 9.2.4 - V924-CD924-L211028
[AUDIT ] CWWKZ0001I: Application com.ibm.mq.rest started in 1.763 seconds.
[AUDIT ] CWWKZ0001I: Application com.ibm.mq.console started in 2.615 seconds.
```

```
[AUDIT ] CWWKF0011I: The mqweb server is ready to run a smarter planet. The mqweb server started in 10.016 seconds.
```

Vous pouvez arrêter le serveur mqweb à tout moment en arrêtant la tâche démarrée du serveur mqweb sous z/OS ou en utilisant 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.

10. z/OS

Facultatif : Sous z/OS, si vous souhaitez autoriser les produits d'automatisation système à intercepter les messages MQWB2019I et MQWB0023I émis lors du démarrage de IBM MQ Console et REST API, configurez le serveur mqweb pour qu'il écrive ces messages sur la console MVS. Pour configurer le serveur mqweb afin qu'il écrive les messages MQWB2019I et MQWB0023I sur la console MVS, éditez le fichier `mqwebuser.xml` que vous avez créé à l'étape «4», à la page 884 et ajoutez la ligne suivante au fichier :

```
<zosLogging enableLogToMVS="true" wtoMessage="MQWB2019I,MQWB0023I" />
```

Pour plus d'informations sur la configuration de la journalisation z/OS dans le serveur mqweb, voir [z/OS Logging \(zosLogging\)](#).

Que faire ensuite

1. Configurez les paramètres du serveur mqweb, notamment l'activation des connexions HTTP et la modification du numéro de port. Pour plus d'informations, voir [«Configuration de IBM MQ Console et de REST API»](#), à la page 883.
2. Si vous le souhaitez, configurez le REST API:
 - a. Configurez 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](#).
 - b. Configurez REST API pour MFT. Pour plus d'informations, voir [«Configuration de REST API pour MFT»](#), à la page 909.
3. Utilisez REST API ou IBM MQ Console:
 - [Initiation à administrative REST API](#)
 - [Initiation à messaging REST API](#)
 - [Initiation à IBM MQ Console](#)

V 9.4.0

Linux

Configuration du IBM MQ Web Server autonome

Depuis IBM MQ 9.4.0, vous pouvez exécuter le serveur mqweb qui héberge IBM MQ Console et REST API dans une installation IBM MQ Web Server autonome.

Avant de commencer

Le IBM MQ Web Server autonome est disponible uniquement sous Linux.

Avant de configurer le serveur mqweb, vous devez installer le IBM MQ Web Server en suivant les étapes de la rubrique [Installation du IBM MQ Web Server autonome](#).

Pourquoi et quand exécuter cette tâche

Suivez la procédure de cette tâche pour créer et configurer un nouveau serveur mqweb qui s'exécute dans une installation IBM MQ Web Server autonome. Vous pouvez configurer plusieurs serveurs mqweb pour qu'ils s'exécutent dans une installation IBM MQ Web Server autonome en répétant cette procédure.

Procédure

1. Créez le répertoire de données IBM MQ Web Server .

Le répertoire de données est utilisé pour stocker les fichiers de configuration et les fichiers journaux du serveur mqweb qui exécute IBM MQ Console et REST API. Vous pouvez utiliser n'importe quel répertoire que vous choisissez comme répertoire de données IBM MQ Web Server .

L'ID utilisateur que vous utilisez pour démarrer le serveur mqweb doit disposer d'un accès en lecture et en écriture au répertoire de données.

2. Définissez la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données que vous avez créé à l'étape «1», à la page 888.

Par exemple, si vous avez choisi d'utiliser `/var/mqweb` comme répertoire de données IBM MQ Web Server , exécutez la commande suivante:

```
export MQ_OVERRIDE_DATA_PATH=/var/mqweb
```

3. Utilisez la commande **setmqenv** pour configurer l'environnement IBM MQ .

Accédez au répertoire `bin` du répertoire d'installation IBM MQ Web Server , puis exécutez la commande suivante:

```
. setmqenv -s
```

4. Utilisez la commande **crtmqdir** pour créer les répertoires et les fichiers IBM MQ dans le répertoire de données. Les fichiers créés incluent une définition de modèle pour le serveur mqweb.

Entrez la commande suivante :

```
crtmqdir -s -f
```

5. Facultatif : Si ce serveur mqweb est le premier que vous avez créé pour s'exécuter avec cette installation du IBM MQ Web Server autonome, utilisez la commande **mqlicense** pour vérifier et accepter la licence IBM MQ .

Vous devez exécuter cette commande en tant qu'utilisateur disposant d'un accès en écriture au répertoire d'installation d' IBM MQ Web Server .

Par exemple, exécutez la commande suivante pour afficher la licence IBM MQ :

```
mqlicense
```

Pour plus d'informations, voir [mqlicense](#).

6. Facultatif : Pour migrer un serveur mqweb existant à exécuter dans l'installation IBM MQ Web Server autonome nouvellement configurée, procédez comme suit:

- a. Sauvegardez votre configuration de serveur mqweb existante.
- b. Restaurez les fichiers dans le répertoire `MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST` , où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server que vous avez créé à l'étape «1», à la page 888.

Pour plus d'informations, voir «[Sauvegarde et restauration de la configuration de votre serveur mqweb](#)», à la page 918.

Remarque : Certaines fonctions de IBM MQ Console et REST API ne sont pas disponibles dans une installation IBM MQ Web Server autonome. Si vous migrez un serveur mqweb d'une installation IBM MQ vers une installation IBM MQ Web Server autonome, ces fonctions ne peuvent pas être utilisées après la migration. Pour plus d'informations sur les restrictions qui s'appliquent dans une installation IBM MQ Web Server autonome, voir [IBM MQ Console et REST API](#).

Que faire ensuite

Configurez le serveur mqweb en suivant les étapes décrites dans «[Configuration de base pour le serveur mqweb](#)», à la page 883.

Configuration de la sécurité

Vous pouvez configurer la sécurité pour IBM MQ Console et REST API en éditant le fichier `mqwebuser.xml`. Vous pouvez configurer et authentifier les utilisateurs en configurant un registre d'utilisateurs de base, un registre LDAP ou tout autre type de registre fourni avec WebSphere Liberty. Vous pouvez ensuite autoriser ces utilisateurs en leur affectant un rôle.

Pourquoi et quand exécuter cette tâche

Pour configurer la sécurité pour IBM MQ Console et REST API, vous devez configurer les utilisateurs et les groupes. Ces utilisateurs et ces groupes peuvent ensuite être autorisés à utiliser IBM MQ Console et/ou REST API. Pour plus d'informations sur la configuration des utilisateurs et des groupes, ainsi que sur l'authentification et l'autorisation des utilisateurs, voir [Sécurité IBM MQ Console et REST API](#).

Lorsque les utilisateurs s'authentifient auprès de IBM MQ Console, un jeton LTPA est généré. Ce jeton permet à l'utilisateur d'utiliser le IBM MQ Console sans nouvelle authentification jusqu'à l'expiration du jeton.

Si vous utilisez l'authentification basée sur un jeton avec REST API, un jeton LTPA différent est généré lorsque l'utilisateur se connecte à l'aide de la ressource `/login` REST API avec la méthode HTTP POST. Vous pouvez configurer la date d'expiration de ce jeton et indiquer s'il peut être utilisé pour les connexions HTTP et HTTPS. Pour plus d'informations, voir [«Configuration du jeton LTPA»](#), à la page 897.

Procédure





- [Sécurité IBM MQ Console et REST API](#)
- [«Configuration du jeton LTPA»](#), à la page 897

Configuration du nom d'hôte HTTP

Par défaut, le serveur `mqweb` qui héberge IBM MQ Console et REST API est configuré pour autoriser uniquement les connexions locales. C'est-à-dire que IBM MQ Console et REST API sont accessibles uniquement sur le système sur lequel IBM MQ Console et REST API sont installés. Vous pouvez configurer le nom d'hôte pour autoriser les connexions distantes à l'aide de la commande `setmqweb`.

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes `dspmqweb` et `setmqweb` :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes `dspmqweb` et `setmqweb` et d'accéder en écriture au fichier `mqwebuser.xml`.
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
-   Si le serveur `mqweb` fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server.



Avertissement :

Avant d'exécuter les commandes `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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).



Avertissement : Linux V 9.4.0

Avant d'exécuter les commandes **setmqweb** ou **dspmqweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Procédure

- Affichez la configuration en cours du nom d'hôte HTTP à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `httpHost` affiche le nom d'hôte HTTP.

- Définissez le nom d'hôte HTTP à l'aide de la commande suivante:

```
setmqweb properties -k httpHost -v hostName
```

où *hostName* 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 entre guillemets pour spécifier toutes les interfaces de réseau disponibles. Utilisez la valeur `localhost` pour n'autoriser que les connexions locales.

- Annulez la définition du nom d'hôte HTTP à l'aide de la commande suivante:

```
setmqweb properties -k httpHost -d
```




Configuration des ports HTTP et HTTPS

Par défaut, le serveur mqweb qui héberge IBM MQ Console et REST API utilise le port HTTPS 9443. Le port associé aux connexions HTTP est désactivé. Vous pouvez activer le port HTTP, configurer un port HTTPS différent ou désactiver le port HTTP ou HTTPS. Vous pouvez configurer les ports à l'aide de la commande **setmqweb** .

Avant de commencer

Si vous activez le port HTTP et que vous utilisez l'authentification basée sur un jeton, vous devez activer le même jeton LTPA à utiliser pour les connexions HTTP et HTTPS. Pour plus d'informations, voir «[Configuration du jeton LTPA](#)», à la page 897.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#).
-  Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où *WLP_user_directory* est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).



Avertissement : Linux V 9.4.0

Avant d'exécuter les commandes **setmqweb** ou **dspmqweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .



Avertissement : Par défaut, le serveur mqweb requiert que les jetons LTPA soient sécurisés pour toutes les demandes. Si le serveur mqweb est configuré pour exiger la sécurisation des jetons LTPA, vous ne pouvez pas effectuer les actions suivantes lorsque vous vous connectez au port HTTP:

- Connectez-vous à IBM MQ Console.
- Utilisez l'authentification basée sur un jeton avec REST API.

Pour permettre aux jetons LTPA d'être utilisés par les demandes HTTP, définissez la valeur de la propriété **secureLTPA** sur `false`. Pour plus d'informations, voir [«Configuration du jeton LTPA»](#), à la page 897.

Procédure

- Affichez la configuration actuelle des ports HTTP et HTTPS à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `httpPort` indique le port HTTP et la zone `httpsPort` indique le port HTTPS.

- Activez ou configurez le port HTTP à l'aide de la commande suivante:

- Activez ou définissez le port HTTP à l'aide de la commande suivante:

```
setmqweb properties -k httpPort -v portNumber
```

où *portNumber* indique le port que vous souhaitez utiliser pour les connexions HTTP. Vous pouvez désactiver le port en utilisant la valeur `-1`.

- Réinitialisez la valeur du port HTTP à la valeur par défaut `-1` à l'aide de la commande suivante:

```
setmqweb properties -k httpPort -d
```

- Configurez le port HTTPS:

- Définissez le numéro de port HTTPS à l'aide de la commande suivante:

```
setmqweb properties -k httpsPort -v portNumber
```

où *portNumber* indique le port à utiliser pour les connexions HTTPS. Vous pouvez désactiver le port en utilisant la valeur `-1`.

- Réinitialisez le numéro de port HTTPS à la valeur par défaut `9443` à l'aide de la commande suivante:

```
setmqweb properties -k httpsPort -d
```

Configuration du délai d'attente de réponse

Par défaut, IBM MQ Console et REST API dépassent le délai d'attente si le délai d'envoi d'une réponse à un client est supérieur à 30 secondes. Vous pouvez configurer IBM MQ Console et REST API pour qu'ils utilisent une autre valeur de délai d'attente à l'aide de la commande **setmqweb** .

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmweb** et **setmqweb** :

- **z/OS** Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
- **Multi** Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
- **Linux V 9.4.0** Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** 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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).



Avertissement : **Linux V 9.4.0**

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Procédure




- Affichez la configuration en cours du délai d'attente de la demande à l'aide de la commande suivante:
`dspmweb properties -a`
La zone `mqRestRequestTimeout` affiche la valeur en cours du délai de réponse. Pour plus d'informations, voir [Propriétés dspmweb](#).
- Définissez le délai d'attente de la demande à l'aide de la commande suivante:
`setmqweb properties -k mqRestRequestTimeout -v timeout`
où `timeout` indique la durée, en secondes, avant l'expiration du délai.
- Réinitialisez le délai d'attente de la demande à la valeur par défaut de 30 secondes à l'aide de la commande suivante:
`setmqweb properties -k mqRestRequestTimeout -d`

Configuration du démarrage automatique

Par défaut, IBM MQ Console est automatiquement démarré lorsque le serveur mqweb démarre. Vous pouvez configurer le démarrage automatique du IBM MQ Console et du REST API à l'aide de la commande **setmqweb** .

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
-  Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** 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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).



Avertissement :

 **Linux V 9.4.0**

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Procédure

- Affichez la configuration actuelle du démarrage automatique à l'aide de la commande suivante:
`dspmweb properties -a`

La zone `mqRestAutostart` indique si le REST API est démarré automatiquement et la zone `mqConsoleAutostart` indique si le IBM MQ Console est démarré automatiquement.

- Configurez le démarrage automatique de IBM MQ Console à l'aide de la commande suivante:
`setmqweb properties -k mqConsoleAutostart -v start`

où `start` est la valeur `true` si vous souhaitez que IBM MQ Console démarre automatiquement, ou `false` dans le cas contraire.

- Configurez le démarrage automatique de REST API à l'aide de la commande suivante:

```
setmqweb properties -k mqRestAutostart -v start
```




où `start` est la valeur `true` si vous souhaitez que REST API démarre automatiquement, ou `false` dans le cas contraire.

Configuration de la journalisation

Vous pouvez configurer les niveaux de journalisation, la taille maximale des fichiers journaux et le nombre maximal de fichiers journaux utilisés par le serveur mqweb qui héberge IBM MQ Console et REST API. Vous pouvez configurer la journalisation à l'aide de la commande **setmqweb** .

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
-  Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** 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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Pourquoi et quand exécuter cette tâche

Le serveur mqweb écrit les messages de journal et la trace dans les fichiers journaux suivants:

console.log et messages.log


Ces fichiers contiennent les messages émis par le IBM MQ Console, le REST API et le serveur mqweb qui exécute ces composants.


trace.log


Ce fichier contient la trace pour IBM MQ Console et REST API. La trace est écrite dans ce fichier uniquement si la trace est activée.

Les fichiers journaux du serveur mqweb se trouvent dans l'un des répertoires suivants:

- Dans une installation IBM MQ :

–  Sous AIX ou Linux: `/var/mqm/web/installations/installationName/servers/mqweb/logs`

–  Sous Windows:
`MQ_DATA_PATH\web\installations\installationName\servers\mqweb\logs`, où `MQ_DATA_PATH` est le chemin de données IBM MQ . Il s'agit du chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.

–  Sous z/OS : `WLP_user_directory/servers/mqweb/logs`

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

- V 9.4.0
Linux
 Dans une installation IBM MQ Web Server autonome:
`MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST/servers/mqweb/logs`
 où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel la variable d'environnement **MQ_OVERRIDE_DATA_PATH** pointe.

Les fichiers de trace de messagerie pour le code REST API de messagerie qui s'exécute sur le serveur mqweb se trouvent dans l'un des répertoires suivants:

- Dans une installation IBM MQ :
 - Linux
AIX
 Sous AIX ou Linux: `/var/mqm/web/installations/installationName/servers/mqweb`
 - Windows
 Sous Windows:
`MQ_DATA_PATH\web\installations\installationName\servers\mqweb`, où `MQ_DATA_PATH` est le chemin de données IBM MQ . Il s'agit du chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.
 - 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** pour créer la définition de serveur mqweb.
- V 9.4.0
Linux
 Dans une installation IBM MQ Web Server autonome:
`MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST/servers/mqweb`
 où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel la variable d'environnement **MQ_OVERRIDE_DATA_PATH** pointe.

Pour plus d'informations sur l'activation de la trace pour:

- REST API, voir [Traçage du REST API](#)
- IBM MQ Console, voir [Traçage du IBM MQ Console](#)

Procédure

- Affichez la configuration en cours de la journalisation REST API à l'aide de la commande suivante:
`dspmweb properties -a`
 - La zone `maxTraceFileSize` indique la taille maximale du fichier journal
 - La zone `maxTraceFiles` affiche le nombre maximal de fichiers journaux
 - La zone `traceSpec` indique le niveau de trace utilisé
 - La zone `maxMsgTraceFileSize` affiche la taille maximale du fichier de trace de la messagerie
 - La zone `maxMsgTraceFiles` affiche le nombre maximal de fichiers de trace de messagerie
- Configurez la taille maximale des fichiers `messages.log` et `trace.log` :
 - Définissez la taille maximale du fichier journal à l'aide de la commande suivante:
`setmqweb properties -k maxTraceFileSize -v size`
 où `size` indique la taille, en Mo, que chaque fichier journal peut atteindre.
 - Réinitialisez la taille maximale du fichier journal à la valeur par défaut de 20 Mo à l'aide de la commande suivante:
`setmqweb properties -k maxTraceFileSize -d`
- Configurez le nombre maximal de fichiers `messages.log` et `trace.log` :
 - Définissez le nombre maximal de chaque fichier journal à l'aide de la commande suivante:
`setmqweb properties -k maxTraceFiles -v max`

- où *max* indique le nombre maximal de fichiers.
- Réinitialisez le nombre maximal de chaque fichier journal à la valeur par défaut 2 à l'aide de la commande suivante:


```
setmqweb properties -k maxTraceFiles -d
```
 - Configurez la taille maximale du fichier de trace de la messagerie:
 - Définissez la taille maximale du fichier de trace de messagerie à l'aide de la commande suivante:


```
setmqweb properties -k maxMsgTraceFileSize -v size
```

 où *size* indique la taille, en Mo, que chaque fichier de trace de messagerie peut atteindre.
 - Réinitialisez la taille maximale du fichier de trace de messagerie à la valeur par défaut de 200 Mo à l'aide de la commande suivante:


```
setmqweb properties -k maxMsgTraceFileSize -d
```
 - Configurez le nombre maximal de fichiers de trace de messagerie à utiliser:
 - Définissez le nombre maximal de fichiers à utiliser pour la trace de messagerie à l'aide de la commande suivante:


```
setmqweb properties -k maxMsgTraceFiles -v max
```

 où *max* indique le nombre maximal de fichiers.
 - Réinitialisez le nombre maximal de fichiers à utiliser pour la trace de messagerie à la valeur par défaut 5 à l'aide de la commande suivante:


```
setmqweb properties -k maxMsgTraceFiles -d
```
 - Configurez le niveau de trace que le serveur mqweb écrit:
 - Définissez la spécification de trace utilisée à l'aide de la commande suivante:


```
setmqweb properties -k traceSpec -v level
```

 où *niveau* est l'une des valeurs répertoriées dans le [Tableau 52](#), à la page 896. Le tableau présente les niveaux de journalisation, classés par niveau de détail croissant. Lorsque vous activez un niveau de journalisation, vous activez également chaque niveau qui le précède. Par exemple, si vous activez le niveau de journalisation ***=warning**, vous activez également les niveaux de journalisation ***=severe** et ***=fatal**.
 Modifiez cette valeur à la demande du support IBM.
 - Réinitialisez la spécification de trace utilisée à la valeur par défaut ***=info** à l'aide de la commande suivante:


```
setmqweb properties -k traceSpec -d
```

Tableau 52. Niveaux de journalisation valides	
Valeur	Niveau de journalisation appliqué
* = désactivé	La journalisation est désactivée.
* =fatal	La tâche ne peut pas continuer et le composant, l'application et le serveur ne peuvent pas fonctionner.
* =grave	La tâche ne peut pas continuer mais le composant, l'application et le serveur peuvent toujours fonctionner. Ce niveau peut également indiquer une erreur irrémédiable imminente.
* =avertissement	Erreur potentielle ou erreur gênante. Ce niveau peut également indiquer un échec progressif (par exemple, la fuite potentielle de ressources).

Tableau 52. Niveaux de journalisation valides (suite)	
Valeur	Niveau de journalisation appliqué
*=audit	Événement significatif affectant l'état du serveur ou les ressources
*=info	Informations générales décrivant l'état d'avancement global de la tâche
*=config	Changement de configuration ou de statut
*=détail	Informations générales détaillant la progression de la sous-tâche
*=fin	Informations de trace-Trace générale + valeurs d'entrée, de sortie et de retour de la méthode
*=plus précis	Informations de trace-Trace détaillée
*=le plus fin	Informations de trace-Trace plus détaillée incluant tous les détails nécessaires au débogage des problèmes
*=all	Tous les événements sont consignés

Configuration du jeton LTPA

Les jetons LTPA peuvent être utilisés pour éviter d'avoir besoin d'un utilisateur pour fournir des données d'identification par nom d'utilisateur et mot de passe sur chaque demande au serveur mqweb. Vous pouvez configurer le nom du cookie de jeton LTPA, l'intervalle d'expiration des jetons d'authentification LTPA et indiquer si les jetons LTPA peuvent être utilisés par les connexions HTTP à l'aide de la commande **setmqweb**.

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

- ▶ **z/OS** Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml`.
- ▶ **Multi** Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
- ▶ **Linux** **V 9.4.0** Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server.

Remarque : Si vous utilisez à la fois l' IBM MQ Console et l'authentification par jeton avec REST API, l'intervalle d'expiration est partagé.



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).





Avertissement : Linux V 9.4.0


Avant d'exécuter les commandes **setmqweb** ou **dspmqweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Pourquoi et quand exécuter cette tâche

Lorsque les utilisateurs se connectent à IBM MQ Console, un jeton LTPA est généré. Si vous utilisez l'authentification basée sur un jeton avec REST API, un jeton LTPA est généré lorsque l'utilisateur se connecte à l'aide de la ressource `/login` REST API avec la méthode HTTP POST. Ce jeton est renvoyé dans un cookie. Le jeton est utilisé pour authentifier l'utilisateur sans qu'il ne soit tenu de se reconnecter avec son ID utilisateur et son mot de passe, jusqu'à l'expiration du jeton. L'intervalle d'expiration par défaut est de 120 minutes.

Le nom du cookie qui inclut le jeton LTPA varie en fonction de la plateforme:

-  Sur le IBM MQ Appliance, le jeton LTPA est `LtpaToken2`. Cette valeur n'est pas modifiable.
-  Par défaut, sur toutes les autres plateformes, le nom du cookie qui inclut le jeton LTPA commence par `LtpaToken2` et inclut un suffixe qui peut être modifié lorsque le serveur `mqweb` est redémarré. Ce nom de cookie aléatoire permet à plusieurs serveurs `mqweb` de s'exécuter sur le même système. Toutefois, si vous souhaitez que le nom du cookie reste cohérent, vous pouvez spécifier le nom du cookie à l'aide de la commande **setmqweb** .

 Si vous activez les ports HTTP et HTTPS, un jeton LTPA émis pour une demande HTTPS peut être réutilisé pour une demande HTTP. Ce comportement est désactivé par défaut, mais vous pouvez l'activer à l'aide de la commande **setmqweb** .

Procédure

- Affichez l'expiration en cours du jeton LTPA, le nom du cookie de jeton LTPA et indiquez si le jeton LTPA peut être utilisé pour les demandes HTTP à l'aide de la commande suivante:


```
dspmqweb properties -a
```

- La zone `ltpaCookieName` affiche le nom du cookie de jeton LTPA. Si vous n'avez pas défini de nom de cookie, la valeur de cette propriété est `LtpaToken2_${env.MQWEB_LTPA_SUFFIX}` on AIX, Linux, and Windows ou `LtpaToken2_${httpsPort}` on z/OS, . La variable après le préfixe `LtpaToken2_` est utilisée par le serveur `mqweb` pour générer un nom unique pour le cookie. Vous ne pouvez pas définir cette variable, mais vous pouvez remplacer `ltpaCookieName` par une valeur de votre choix.
 - La zone `ltpaExpiration` affiche le délai d'expiration du jeton LTPA.
 - La zone `secureLtpa` est définie sur `false` si des jetons LTPA peuvent être utilisés par des demandes HTTP.
- Configurez l'expiration du jeton LTPA:
 - Définissez l'expiration du jeton LTPA en entrant la commande suivante:

```
setmqweb properties -k ltpaExpiration -v time
```

où `time` indique la durée, en minutes, avant l'expiration du jeton LTPA et la déconnexion de l'utilisateur.
 - Réinitialisez l'expiration du jeton LTPA à la valeur par défaut de 120 minutes en entrant la commande suivante:

```
setmqweb properties -k ltpaExpiration -d
```


- 

Configurez le nom du cookie de jeton LTPA:

 - Définissez le nom du cookie de jeton LTPA en entrant la commande suivante:


```
setmqweb properties -k ltpaCookieName -v name
```

 où *name* indique un nom unique pour le cookie de jeton LTPA.
 - Réinitialisez le nom du cookie de jeton LTPA à la valeur par défaut, où le préfixe `LtpaToken2_` est suivi de caractères aléatoires, en entrant la commande suivante:


```
setmqweb properties -k ltpaCookieName -d
```
- 

Indiquez si le jeton LTPA peut être utilisé par les connexions HTTP en entrant la commande suivante:


```
setmqweb properties -k secureLtpa -v secure
```




 où *secure* indique si le jeton LTPA peut être utilisé par des connexions HTTP non sécurisées et des connexions HTTPS sécurisées. La valeur *false* permet aux connexions HTTP et HTTPS d'utiliser le même jeton LTPA.

Configuration du comportement de connexion du gestionnaire de files d'attente éloignées pour IBM MQ Console

Lorsque vous utilisez IBM MQ Console, vous pouvez créer des connexions à des gestionnaires de files d'attente éloignées. Autrement dit, vous pouvez vous connecter à des gestionnaires de files d'attente qui ne font pas partie de la même installation que le serveur `mqweb` qui exécute le IBM MQ Console. Il existe un certain nombre d'options de configuration que vous pouvez définir pour contrôler le comportement des connexions de gestionnaire de files d'attente éloignées.

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes `dspmweb` et `setmqweb` :

- 
 Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes `dspmweb` et `setmqweb` et d'accéder en écriture au fichier `mqwebuser.xml` .
- 
 Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#).
- 
 Si le serveur `mqweb` fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes `setmqweb` ou `dspmweb` 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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmqweb** dans une installation IBM MQ Web Server autonome, vous devez définir la variable d'environnement **MQ_OVERRIDE_DATA_PATH** sur le répertoire de données IBM MQ Web Server .

Pourquoi et quand exécuter cette tâche

Vous pouvez définir les options de configuration suivantes:

- Indique si les connexions de gestionnaire de files d'attente éloignées sont autorisées.
- Indique si les connexions peuvent être ajoutées à l'aide de la IBM MQ Console ou uniquement à l'aide de la ligne de commande.
- Indique si les gestionnaires de files d'attente locaux sont affichés dans IBM MQ Console lorsque des connexions de gestionnaire de files d'attente éloignées sont autorisées.
- Indique si les connexions de gestionnaire de files d'attente éloignées sont automatiquement établies lorsque le IBM MQ Console est démarré ou en cas d'échec de la connexion.
- Durée entre chaque actualisation de la liste des gestionnaires de files d'attente éloignées affichés dans le IBM MQ Console.

Procédure

- Pour afficher les paramètres de configuration de la connexion du gestionnaire de files d'attente éloignées en cours, entrez la commande suivante:



```
dspmqweb properties -a
```

- La zone `mqConsoleRemoteSupportEnabled` indique si les connexions de gestionnaire de files d'attente éloignées sont autorisées.
- La zone `mqConsoleRemoteUIAdmin` indique si des connexions de gestionnaire de files d'attente éloignées peuvent être ajoutées à l'aide de la IBM MQ Console.
- La zone `mqConsoleRemoteAllowLocal` indique si les gestionnaires de files d'attente locaux sont affichés.
- La zone `mqConsoleRemotePollTime` indique le nombre de secondes entre chaque actualisation de la liste des gestionnaires de files d'attente éloignées.

- Pour empêcher ou autoriser les connexions de gestionnaire de files d'attente éloignées avec IBM MQ Console, entrez la commande suivante:

```
setmqweb properties -k mqConsoleRemoteSupportEnabled -v true or false
```

où `true` autorise les connexions de gestionnaire de files d'attente éloignées ou `false` empêche les connexions de gestionnaire de files d'attente éloignées.

Remarque :   Si le serveur mqweb s'exécute dans une installation IBM MQ Web Server autonome, la propriété **mqConsoleRemoteSupportEnabled** n'est pas valide. Le IBM MQ Web Server autonome prend en charge uniquement les connexions aux gestionnaires de files d'attente éloignées.

- Pour empêcher ou autoriser l'ajout de connexions de gestionnaire de files d'attente éloignées à l'aide de la IBM MQ Console ou uniquement à l'aide de la ligne de commande, entrez la commande suivante:



```
setmqweb properties -k mqConsoleRemoteUIAdmin -v true or false
```

où `true` permet d'ajouter des connexions de gestionnaire de files d'attente éloignées à l'aide de la IBM MQ Console et de la ligne de commande, ou `false` permet d'ajouter des connexions de gestionnaire de files d'attente éloignées uniquement à l'aide de la commande **setmqweb remote** sur la ligne de commande.

- Pour empêcher ou autoriser l'affichage des gestionnaires de files d'attente locales dans IBM MQ Console lorsque des connexions de gestionnaire de files d'attente éloignées sont autorisées, entrez la commande suivante:

```
setmqweb properties -k mqConsoleRemoteAllowLocal -v true or false
```

où `true` permet d'afficher les gestionnaires de files d'attente locaux ou `false` masque les gestionnaires de files d'attente locaux.


Remarque :   Si le serveur mqweb s'exécute dans une installation IBM MQ Web Server autonome, la propriété `mqConsoleRemoteAllowLocal` n'est pas valide. Le IBM MQ Web Server autonome prend en charge uniquement les connexions aux gestionnaires de files d'attente éloignées.

- Pour définir la durée entre chaque actualisation de la liste des gestionnaires de files d'attente éloignées affichés dans le IBM MQ Console, entrez la commande suivante:

```
setmqweb properties -k mqConsoleRemotePollTime -v seconds
```

où `seconds` est défini sur une valeur entière du nombre de secondes entre chaque actualisation de la liste des gestionnaires de files d'attente éloignées.



Référence associée

 [ID de processus setmqweb dspmqweb](#)



Configuration de la passerelle administrative REST API

Lorsque la passerelle administrative REST API est activée, vous pouvez effectuer une administration à distance avec REST API à l'aide d'un gestionnaire de files d'attente de passerelle. Vous pouvez configurer le gestionnaire de files d'attente utilisé comme gestionnaire de files d'attente de passerelle par défaut ou empêcher l'administration à distance en désactivant la passerelle administrative REST API à l'aide de la commande `setmqweb`.

Avant de commencer

Remarque :   Si le serveur mqweb est exécuté dans une installation IBM MQ Web Server autonome, cette tâche n'est pas applicable. administrative REST API n'est pas disponible dans une installation IBM MQ Web Server autonome.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes `dspmqweb` et `setmqweb` :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes `dspmqweb` et `setmqweb` et d'accéder en écriture au fichier `mqwebuser.xml`.
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.



Avertissement :

Avant d'exécuter les commandes `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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Pourquoi et quand exécuter cette tâche

Lorsque le serveur mqweb s'exécute dans une installation IBM MQ, la passerelle administrative REST API est activée par défaut.

Le gestionnaire de files d'attente de passerelle par défaut 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.

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

Procédure

- Affichez la configuration actuelle de la passerelle administrative REST API à l'aide de la commande suivante:

```
dspmweb properties -a
```

La zone `mqRestGatewayEnabled` indique si la passerelle est activée et la zone `mqRestGatewayQmgr` indique le nom du gestionnaire de files d'attente de passerelle par défaut.

- Configurez si la passerelle administrative REST API est activée à l'aide de la commande suivante:

```
setmqweb properties -k mqRestGatewayEnabled -v enabled
```

où *enabled* est la valeur **true** pour activer la passerelle administrative REST API ou **false** dans le cas contraire.

- Configurez le gestionnaire de files d'attente utilisé comme gestionnaire de files d'attente de passerelle par défaut:

- Définissez le gestionnaire de files d'attente de passerelle par défaut à l'aide de la commande suivante:

```
setmqweb properties -k mqRestGatewayQmgr -v qmgrName
```

où *qmgrName* est le nom d'un gestionnaire de files d'attente dans la même installation que le serveur mqweb.

- Annulez la définition du gestionnaire de files d'attente de passerelle par défaut à l'aide de la commande suivante:

```
setmqweb properties -k mqRestGatewayQmgr -d
```

Configuration de messaging REST API

Vous pouvez configurer le messaging REST API de plusieurs manières. Vous pouvez choisir d'activer ou de désactiver la fonction messaging REST API. Vous pouvez choisir le nombre maximal de connexions en pool pouvant être utilisées par messaging REST API, ainsi que le comportement de messaging REST API lorsque toutes les connexions sont utilisées. Vous pouvez également choisir le contexte utilisateur utilisé pour l'autorisation lorsque vous utilisez messaging REST API pour envoyer, recevoir, parcourir ou publier un message.

Procédure

- [«Activation de messaging REST API», à la page 903](#)
- [«Configuration du regroupement de connexions pour messaging REST API», à la page 904](#)





- 

Activation de messaging REST API

Vous pouvez indiquer si messaging REST API est activé à l'aide de la commande **setmqweb** . Par défaut, messaging REST API est activé.

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
-   Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Procédure

- Affichez la configuration en cours du messaging REST API à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `mqRestMessagingEnabled` indique si messaging REST API est activé. Si la valeur est `True` , messaging REST API est activé.

- Activez le messaging REST API à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingEnabled -v true
```

- Désactivez le messaging REST API à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingEnabled -v false
```

Tâches associées

«Configuration du regroupement de connexions pour messaging REST API», à la page 904

Vous pouvez configurer le nombre maximal de connexions en pool pouvant être utilisées par messaging REST API et le comportement de messaging REST API lorsque toutes les connexions sont utilisées.

«Configuration du contexte utilisateur utilisé pour l'autorisation dans messaging REST API», à la page 907

V 9.4.0 Vous pouvez configurer le contexte utilisateur utilisé pour l'autorisation lorsque vous utilisez messaging REST API pour envoyer, recevoir, parcourir ou publier un message. En d'autres termes, vous pouvez choisir si l'utilisateur qui est connecté à messaging REST API, ou l'utilisateur qui a démarré le serveur mqweb, est utilisé pour l'autorisation.

«Configuration du mode de connexion pour le messaging REST API», à la page 906

Vous pouvez configurer le messaging REST API pour qu'il se connecte à des gestionnaires de files d'attente locaux ou à des gestionnaires de files d'attente éloignées.

Configuration du regroupement de connexions pour messaging REST API

Vous pouvez configurer le nombre maximal de connexions en pool pouvant être utilisées par messaging REST API et le comportement de messaging REST API lorsque toutes les connexions sont utilisées.

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmweb** et **setmqweb** :

- **z/OS** Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
- **Multi** Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.
- **Linux** **V 9.4.0** Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server .



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** 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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Pourquoi et quand exécuter cette tâche

Pour optimiser les performances de l'messaging REST API, les connexions aux gestionnaires de files d'attente IBM MQ sont regroupées en pool. C'est-à-dire qu'au lieu que chaque demande REST crée, utilise et supprime sa propre connexion, chaque demande REST utilise une connexion provenant d'un pool de connexions. Par défaut, 20 connexions sont disponibles pour chaque pool de gestionnaires de files d'attente et vous pouvez choisir l'une des trois options de gestion des demandes lorsque toutes les connexions sont utilisées:

- Le messaging REST API peut créer une nouvelle connexion non regroupée en pool à utiliser pour la demande. Ce comportement est le comportement par défaut.
- Le messaging REST API peut renvoyer une erreur.
- Le messaging REST API peut attendre qu'une connexion en pool soit disponible. Cette attente est indéfinie.

Vous pouvez changer le nombre maximal de connexions en pool et le comportement par défaut de l'messaging REST API lorsque toutes les connexions sont utilisées avec la commande **setmqweb properties**.

Procédure

- Affichez la configuration en cours à l'aide de la commande suivante:

```
dspmqweb properties -a
```

- La zone `mqRestMessagingFullPoolBehavior` indique le comportement du messaging REST API lorsque toutes les connexions du pool sont utilisées. Si la valeur est `block`, messaging REST API doit attendre qu'une connexion soit disponible. Si la valeur est `error`, messaging REST API doit renvoyer une erreur. Si la valeur est `overflow`, messaging REST API doit créer une connexion non regroupée en pool à utiliser et mettre fin à la connexion après utilisation.
- La zone `mqRestMessagingMaxPoolSize` affiche la taille maximale du pool de connexions.
- Configurez le comportement du messaging REST API lorsque toutes les connexions du pool sont utilisées à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingFullPoolBehavior -v action
```

où *action* indique l'action à effectuer. *action* peut avoir l'une des valeurs suivantes:

séquence d'opérations

Lorsque toutes les connexions du pool sont en cours d'utilisation, attendez qu'une connexion soit disponible.

erreur

Lorsque toutes les connexions du pool sont en cours d'utilisation, une erreur est renvoyée.

dépassement

Lorsque toutes les connexions du pool sont en cours d'utilisation, créez une connexion non regroupée en pool à utiliser et éliminez la connexion une fois qu'elle est utilisée.

- Configurez la taille maximale du pool de connexions pour chaque pool de gestionnaires de files d'attente à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingMaxPoolSize -v size
```

où *size* indique la taille du pool.

Remarque : Si une valeur élevée est définie pour `mqRestMessagingMaxPoolSize` et que de nombreux gestionnaires de files d'attente sont connectés, augmentez la taille maximale du segment de mémoire du serveur mqweb. Pour plus d'informations, voir [Optimisation de la machine virtuelle Java du serveur mqweb](#).

Tâches associées

«[Activation de messaging REST API](#)», à la page 903

Vous pouvez indiquer si messaging REST API est activé à l'aide de la commande **setmqweb**. Par défaut, messaging REST API est activé.

«[Configuration du contexte utilisateur utilisé pour l'autorisation dans messaging REST API](#)», à la page 907


V 9.4.0 Vous pouvez configurer le contexte utilisateur utilisé pour l'autorisation lorsque vous utilisez messaging REST API pour envoyer, recevoir, parcourir ou publier un message. En d'autres termes, vous pouvez choisir si l'utilisateur qui est connecté à messaging REST API, ou l'utilisateur qui a démarré le serveur mqweb, est utilisé pour l'autorisation.

«[Configuration du mode de connexion pour le messaging REST API](#)», à la page 906



Vous pouvez configurer le messaging REST API pour qu'il se connecte à des gestionnaires de files d'attente locaux ou à des gestionnaires de files d'attente éloignées.

Vous pouvez configurer le messaging REST API pour qu'il se connecte à des gestionnaires de files d'attente locaux ou à des gestionnaires de files d'attente éloignées.

Avant de commencer

Remarque :  Si le serveur mqweb s'exécute dans une installation IBM MQ Web Server autonome, cette tâche n'est pas applicable. Le IBM MQ Web Server autonome prend en charge uniquement les connexions aux gestionnaires de files d'attente éloignées.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.



Avertissement :

Avant d'exécuter les commandes **setmqweb** ou **dspmweb** 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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```


où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Pourquoi et quand exécuter cette tâche

Le mode de connexion par défaut du messaging REST API varie en fonction du type d'installation qui exécute le serveur mqweb:

- Dans une installation IBM MQ , par défaut, messaging REST API se connecte uniquement aux gestionnaires de files d'attente locaux dans la même installation que le serveur mqweb. Effectuez les étapes de cette tâche pour afficher et modifier la configuration de la connexion.
-  Dans une installation IBM MQ Web Server autonome, messaging REST API prend en charge uniquement les connexions aux gestionnaires de files d'attente éloignées. La configuration de connexion ne peut pas être affichée ou modifiée.

Procédure

- Affichez la configuration en cours du messaging REST API à l'aide de la commande suivante:

```
dspmweb properties -a
```

La zone `mqRestMessagingConnectionMode` affiche le mode de connexion en cours. Si la valeur est `local`, messaging REST API ne peut se connecter qu'aux gestionnaires de files d'attente de la même installation que le serveur mqweb. Si la valeur est `remote`, messaging REST API peut se connecter aux gestionnaires de files d'attente éloignées.

- Configurez le serveur mqweb pour permettre à messaging REST API de se connecter uniquement aux gestionnaires de files d'attente qui se trouvent dans la même installation que le serveur mqweb à l'aide des commandes suivantes:

```
setmqweb properties -k mqRestMessagingConnectionMode -v local
endmqweb
strmqweb
```

- Configurez le serveur mqweb pour permettre à messaging REST API de se connecter aux gestionnaires de files d'attente éloignées à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingConnectionMode -v remote
endmqweb
strmqweb
```

Que faire ensuite

Si vous configurez le serveur mqweb pour permettre à messaging REST API de se connecter à des gestionnaires de files d'attente éloignées, vous devez fournir des informations de connexion pour chaque gestionnaire de files d'attente auquel vous souhaitez vous connecter. Pour plus d'informations sur la manière de fournir les informations de connexion, voir [Configuration d'un gestionnaire de files d'attente éloignées à utiliser avec messaging REST API](#).

Tâches associées

«Activation de messaging REST API», à la page 903

Vous pouvez indiquer si messaging REST API est activé à l'aide de la commande **setmqweb**. Par défaut, messaging REST API est activé.

«Configuration du regroupement de connexions pour messaging REST API», à la page 904

Vous pouvez configurer le nombre maximal de connexions en pool pouvant être utilisées par messaging REST API et le comportement de messaging REST API lorsque toutes les connexions sont utilisées.

«Configuration du contexte utilisateur utilisé pour l'autorisation dans messaging REST API», à la page 907

V 9.4.0 Vous pouvez configurer le contexte utilisateur utilisé pour l'autorisation lorsque vous utilisez messaging REST API pour envoyer, recevoir, parcourir ou publier un message. En d'autres termes, vous pouvez choisir si l'utilisateur qui est connecté à messaging REST API, ou l'utilisateur qui a démarré le serveur mqweb, est utilisé pour l'autorisation.

V 9.4.0 Configuration du contexte utilisateur utilisé pour l'autorisation dans messaging REST API

V 9.4.0 Vous pouvez configurer le contexte utilisateur utilisé pour l'autorisation lorsque vous utilisez messaging REST API pour envoyer, recevoir, parcourir ou publier un message. En d'autres termes, vous pouvez choisir si l'utilisateur qui est connecté à messaging REST API, ou l'utilisateur qui a démarré le serveur mqweb, est utilisé pour l'autorisation.

Avant de commencer

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

- **z/OS** Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml`.
- **Multi** Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#).
- **Linux** **V 9.4.0** Si le serveur mqweb fait partie d'une installation IBM MQ Web Server autonome, vous devez disposer d'un accès en écriture au fichier `mqwebuser.xml` dans le répertoire de données IBM MQ Web Server.



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement WLP_USER_DIR, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où *WLP_user_directory* est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Pourquoi et quand exécuter cette tâche

- Lorsque l'ID utilisateur utilisé est l'ID utilisateur qui est connecté à messaging REST API, **MQMD.UserIdentifier** est défini sur l'ID utilisateur qui est connecté à l'API REST. **MQMD.AppIdentityData** est défini sur l'ID utilisateur qui est connecté à l'API REST.
- Lorsque l'ID utilisateur utilisé est l'ID utilisateur qui a démarré le serveur mqweb, la zone **MQMD.UserIdentifier** est laissée vide. **MQMD.AppIdentityData** est défini sur l'ID utilisateur qui est connecté à l'API REST.

Pour plus d'informations sur les parties du descripteur de message du message IBM MQ , voir [MQMD](#) .

Procédure

- Affichez la configuration en cours du messaging REST API à l'aide de la commande suivante:

```
dspmweb properties -a
```

La zone `mqRestMessagingAdoptWebUserContext` indique l'ID utilisateur utilisé pour l'autorisation lorsque vous envoyez, publiez, recevez ou parcourez des messages. Si la valeur est `True`, l'utilisateur connecté à messaging REST API est utilisé pour l'autorisation. Si la valeur est `False`, l'utilisateur qui a démarré le serveur mqweb est utilisé pour l'autorisation.

- Configurez le messaging REST API pour qu'il utilise l'ID utilisateur de l'utilisateur connecté à messaging REST API pour l'autorisation à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingAdoptWebUserContext -v true
```

Lorsque `mqRestMessagingAdoptWebUserContext` est défini sur `true`, **MQMD.UserIdentifier** est défini sur l'ID utilisateur qui est connecté à l'API REST. **MQMD.AppIdentityData** est défini sur l'ID utilisateur qui est connecté à l'API REST.

- Configurez messaging REST API pour utiliser l'ID utilisateur de l'utilisateur qui a démarré le serveur mqweb à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMessagingAdoptWebUserContext -v false
```

Lorsque `mqRestMessagingAdoptWebUserContext` est défini sur `false`, **MQMD.UserIdentifier** est laissé vide. **MQMD.AppIdentityData** est défini sur l'ID utilisateur qui est connecté à l'API REST.

Tâches associées

«Activation de messaging REST API», à la page 903

Vous pouvez indiquer si messaging REST API est activé à l'aide de la commande `setmqweb` . Par défaut, messaging REST API est activé.

«Configuration du regroupement de connexions pour messaging REST API», à la page 904

Vous pouvez configurer le nombre maximal de connexions en pool pouvant être utilisées par messaging REST API et le comportement de messaging REST API lorsque toutes les connexions sont utilisées.

«Configuration du mode de connexion pour le messaging REST API», à la page 906

Vous pouvez configurer le messaging REST API pour qu'il se connecte à des gestionnaires de files d'attente locaux ou à des gestionnaires de files d'attente éloignées.

Configuration de REST API pour MFT

Par défaut, REST API for MFT n'est pas activé. Vous pouvez indiquer si REST API for MFT est activé, définir le gestionnaire de files d'attente de coordination, définir le gestionnaire de files d'attente de commandes et spécifier le délai d'attente de reconnexion MFT à l'aide de la commande **setmqweb properties**.



Procédure

- «[Activation de REST API pour MFT](#)», à la page 909
- «[Configuration du gestionnaire de files d'attente de coordination pour REST API for MFT](#)», à la page 910
- «[Configuration du gestionnaire de files d'attente de commandes pour REST API for MFT](#)», à la page 911
- «[Configuration des valeurs de délai d'attente REST API for MFT](#)», à la page 912



Activation de REST API pour MFT

Avant de pouvoir utiliser REST API for MFT, vous devez d'abord activer REST API for MFT. Vous pouvez indiquer si REST API for MFT est activé à l'aide de la commande **setmqweb**. Par défaut, REST API for MFT n'est pas activé.

Avant de commencer

Remarque :   Si le serveur mqweb s'exécute dans une installation IBM MQ Web Server autonome, cette tâche n'est pas applicable. REST API for MFT n'est pas disponible dans une installation IBM MQ Web Server autonome.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml`.
-  Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#).



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Procédure

1. Affichez la configuration actuelle de REST API for MFT à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `mqRestMftEnabled` indique si REST API for MFT est activé. La valeur est `True` si REST API for MFT est activé ou `False` dans le cas contraire.

2. Activez ou désactivez REST API for MFT à l'aide de l'une des commandes suivantes:

- Activez REST API for MFT à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMftEnabled -v true
```

- Désactivez REST API for MFT à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMftEnabled -v false
```

3. Redémarrez le serveur mqweb en entrant les commandes suivantes:

```
endmqweb  
strmqweb
```



Que faire ensuite

Si vous avez activé REST API pour MFT, vous devez définir le nom du gestionnaire de files d'attente de coordination avant de pouvoir utiliser REST API pour MFT. Pour plus d'informations sur la définition du gestionnaire de files d'attente de coordination, voir [«Configuration du gestionnaire de files d'attente de coordination pour REST API for MFT»](#), à la page 910.



Configuration du gestionnaire de files d'attente de coordination pour REST API for MFT

Avant de pouvoir utiliser REST API for MFT, vous devez configurer un gestionnaire de files d'attente pour qu'il agisse en tant que gestionnaire de files d'attente de coordination pour les transactions MFT . Vous pouvez définir le gestionnaire de files d'attente qui est le gestionnaire de files d'attente de coordination à l'aide de la commande **setmqweb** .

Avant de commencer

Remarque :   Si le serveur mqweb s'exécute dans une installation IBM MQ Web Server autonome, cette tâche n'est pas applicable. REST API for MFT n'est pas disponible dans une installation IBM MQ Web Server autonome.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#).



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Procédure

1. Affichez la configuration actuelle de REST API for MFT à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `mqRestMftCoordinationQmgr` affiche le nom du gestionnaire de file d'attente de coordination.

2. Configurez le gestionnaire de files d'attente de coordination à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMftCoordinationQmgr -v qmgrName
```

où `qmgrName` est le nom du gestionnaire de files d'attente de coordination. Le gestionnaire de files d'attente de coordination doit se trouver sur la machine sur laquelle le serveur mqweb est en cours d'exécution. Par défaut, ce nom de gestionnaire de files d'attente est vide. Si aucune valeur n'est définie, REST API for MFT ne fonctionne pas.

3. Redémarrez le serveur mqweb en entrant les commandes suivantes:

```
endmqweb  
strmqweb
```



Que faire ensuite

- Vérifiez que REST API for MFT est activé. Pour plus d'informations, voir [«Activation de REST API pour MFT»](#), à la page 909.
- Si vous souhaitez utiliser REST API for MFT pour soumettre des demandes de création, vous devez définir le nom du gestionnaire de files d'attente de commandes. Par exemple, si vous souhaitez utiliser une commande REST API telle que **create transfer**, vous devez définir le nom du gestionnaire de files d'attente de commandes. Pour plus d'informations, voir [«Configuration du gestionnaire de files d'attente de commandes pour REST API for MFT»](#), à la page 911.
- Vous pouvez configurer les valeurs de délai d'attente REST API for MFT . Le délai d'attente par défaut est de 30 minutes. Pour plus d'informations, voir [«Configuration des valeurs de délai d'attente REST API for MFT»](#), à la page 912.
- Pour utiliser REST API for MFT, un utilisateur doit être authentifié auprès du serveur mqweb et être membre d'un ou de plusieurs rôles `MFTWebAdmin` ou `MFTWebAdminRO` . Pour plus d'informations sur la configuration des utilisateurs, voir [Configuration des utilisateurs et des rôles pour REST API](#).



Configuration du gestionnaire de files d'attente de commandes pour REST API for MFT

Avant de pouvoir utiliser REST API for MFT pour soumettre des demandes de création, vous devez définir le nom du gestionnaire de files d'attente de commandes. Par exemple, pour utiliser la ressource **create transfer** , vous devez définir le nom du gestionnaire de files d'attente de commandes. Vous pouvez définir le nom du gestionnaire de files d'attente de commandes à l'aide de la commande **setmqweb** .

Avant de commencer

Remarque :   Si le serveur mqweb s'exécute dans une installation IBM MQ Web Server autonome, cette tâche n'est pas applicable. REST API for MFT n'est pas disponible dans une installation IBM MQ Web Server autonome.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un [utilisateur privilégié](#).



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement WLP_USER_DIR, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où *WLP_user_directory* est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Procédure

1. Affichez la configuration actuelle de REST API for MFT à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `mqRestMftCommandQmgr` affiche le nom du gestionnaire de files d'attente de commandes.

2. Configurez le gestionnaire de files d'attente de commandes à l'aide de la commande suivante:

```
setmqweb properties -k mqRestMftCommandQmgr -v qmgrName
```

où *qmgrName* est le nom du gestionnaire de files d'attente de commandes. Le gestionnaire de files d'attente de commandes doit se trouver sur la machine sur laquelle le serveur mqweb s'exécute. Par défaut, ce nom de gestionnaire de files d'attente est vide. Si une valeur n'est pas définie, REST API for MFT pour une commande de création ne fonctionne pas.

3. Redémarrez le serveur mqweb en entrant les commandes suivantes:

```
endmqweb  
strmqweb
```



Que faire ensuite

- Vérifiez que REST API for MFT est activé. Pour plus d'informations, voir [«Activation de REST API pour MFT»](#), à la page 909.
- Vérifiez qu'un gestionnaire de files d'attente de coordination est défini. Pour plus d'informations, voir [«Configuration du gestionnaire de files d'attente de coordination pour REST API for MFT»](#), à la page 910.
- Vous pouvez configurer les valeurs de délai d'attente REST API for MFT . Le délai d'attente par défaut est de 30 minutes. Pour plus d'informations, voir [«Configuration des valeurs de délai d'attente REST API for MFT»](#), à la page 912.
- Pour utiliser REST API for MFT, un utilisateur doit être authentifié auprès du serveur mqweb et être membre d'un ou de plusieurs rôles `MFTWebAdmin` ou `MFTWebAdminRO` . Pour plus d'informations sur la configuration des utilisateurs, voir [Configuration des utilisateurs et des rôles pour REST API](#).



Configuration des valeurs de délai d'attente REST API for MFT

Vous pouvez configurer la durée, en minutes, après laquelle REST API for MFT cesse de tenter de se connecter au gestionnaire de file d'attente de coordination après l'interruption de la connexion. Le délai d'attente par défaut est de 30 minutes. Vous pouvez configurer ce délai d'attente à l'aide de la commande **setmqweb** .

Avant de commencer

Remarque :   Si le serveur mqweb est exécuté dans une installation IBM MQ Web Server autonome, cette tâche n'est pas applicable. REST API for MFT n'est pas disponible dans une installation IBM MQ Web Server autonome.

Pour effectuer cette tâche, vous devez être un utilisateur disposant de certains privilèges pour pouvoir utiliser les commandes **dspmqweb** et **setmqweb** :

-  Sous z/OS, vous devez disposer des droits permettant d'exécuter les commandes **dspmqweb** et **setmqweb** et d'accéder en écriture au fichier `mqwebuser.xml` .
-  Sur tous les autres systèmes d'exploitation, vous devez être un utilisateur privilégié.



Avertissement :

Avant d'exécuter les commandes **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 définir la variable d'environnement `WLP_USER_DIR`, exécutez la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à `crtmqweb`. Exemple :

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

Pour plus d'informations, voir [Création du serveur mqweb](#).

Pourquoi et quand exécuter cette tâche

Vous pouvez configurer le délai d'attente pour REST API for MFT.

REST API for MFT tente de rétablir la connexion immédiatement après l'interruption de la connexion au gestionnaire de file d'attente de coordination. Si cette tentative échoue, il y a un intervalle de cinq minutes entre chaque tentative de reconnexion, jusqu'à ce que le délai d'attente soit dépassé. Par conséquent, si vous définissez une valeur comprise entre 0 et 5, une seule tentative de reconnexion est effectuée.

Une fois la reconnexion arrivée à expiration, la prochaine tentative de reconnexion est effectuée lorsque l'une des ressources REST API for MFT est appelée. Si cette tentative de reconnexion échoue, MFT tente à nouveau de se reconnecter toutes les cinq minutes jusqu'à ce que le délai de reconnexion soit écoulé.

Procédure

1. Affichez la configuration actuelle de REST API for MFT à l'aide de la commande suivante:

```
dspmqweb properties -a
```

La zone `mqRestMftReconnectTimeoutInMinutes` affiche la valeur du délai de reconnexion, jusqu'à ce que les services MFT Transfer Rest cessent de tenter de se connecter au gestionnaire de files d'attente de coordination.

2. Configurez le délai d'attente, en minutes, après lequel REST API for MFT cesse de tenter de se connecter au gestionnaire de files d'attente de coordination:

- Réinitialisez le délai d'attente à la valeur par défaut de 30 minutes:

```
setmqweb properties -k mqRestMftReconnectTimeoutInMinutes -d
```

- Définissez le délai d'attente:

```
setmqweb properties -k mqRestMftReconnectTimeoutInMinutes -v time
```

où *time* indique la durée, en minutes, avant le dépassement du délai d'attente.

Si cette valeur est définie entre 0-5, REST API for MFT tente de se reconnecter au gestionnaire de files d'attente de coordination une seule fois. Si la connexion échoue, aucune tentative n'est tentée pour rétablir la connexion tant que le REST API n'est pas appelé.

Si cette valeur est définie sur -1, REST API for MFT tente de se reconnecter jusqu'à ce que la connexion aboutisse.

3. Redémarrez le serveur mqweb en entrant les commandes suivantes:

```
endmqweb  
startmqweb
```

Que faire ensuite

- Vérifiez que REST API for MFT est activé. Pour plus d'informations, voir [«Activation de REST API pour MFT»](#), à la page 909.
- Vérifiez qu'un gestionnaire de files d'attente de coordination est défini. Pour plus d'informations, voir [«Configuration du gestionnaire de files d'attente de coordination pour REST API for MFT»](#), à la page 910.
- Si vous souhaitez utiliser REST API for MFT pour soumettre des demandes de création, vous devez définir le nom du gestionnaire de files d'attente de commandes. Par exemple, si vous souhaitez utiliser une commande REST API telle que **create transfer**, vous devez définir le nom du gestionnaire de files d'attente de commandes. Pour plus d'informations, voir [«Configuration du gestionnaire de files d'attente de commandes pour REST API for MFT»](#), à la page 911.
- Pour utiliser REST API for MFT, un utilisateur doit être authentifié auprès du serveur mqweb et être membre d'un ou de plusieurs rôles MFTWebAdmin ou MFTWebAdminRO . Pour plus d'informations sur la configuration des utilisateurs, voir [Configuration des utilisateurs et des rôles pour REST API](#).

Optimisation de la machine virtuelle Java du serveur mqweb

Par défaut, le serveur mqweb Java Virtual Machine (JVM) utilise des valeurs par défaut spécifiques à la plateforme pour les paramètres de configuration, tels que la taille minimale et maximale du segment de mémoire et la taille du cache de classe.

Pourquoi et quand exécuter cette tâche

Vous devrez peut-être modifier les valeurs par défaut pour améliorer les performances ou résoudre les problèmes. Par exemple, si un `java.lang.OutOfMemoryError` est émis par le serveur mqweb, vous devez augmenter la taille maximale du segment de mémoire. Vous devez également augmenter la taille du segment de mémoire si vous tentez de charger un grand nombre d'objets de file d'attente.


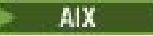

Si vous rencontrez des problèmes avec l'affichage des informations de configuration de tableau de bord dans le IBM MQ Console, vous devez définir une variable qui détermine le codage de fichier de la configuration. Vous pouvez modifier les valeurs par défaut dans le fichier `jvm.options` .

Procédure


1. Ouvrez le fichier `jvm.options`.

Le fichier `jvm.options` se trouve dans l'un des répertoires suivants:

- Dans une installation IBM MQ :

-   Sous AIX ou Linux: `/var/mqm/web/installations/installationName/servers/mqweb`
-  Sous Windows:
`MQ_DATA_PATH\web\installations\installationName\servers\mqweb`, où `MQ_DATA_PATH` est le chemin de données IBM MQ . Il s'agit du chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.

–  Sous IBM i : `MQ_DATA_PATH/web/installations/Installation1/`

–  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` pour créer la définition de serveur mqweb.

-   Dans une installation IBM MQ Web Server autonome :
`MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST/servers/mqweb`
où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel la variable d'environnement `MQ_OVERRIDE_DATA_PATH` pointe.

2. Facultatif : Définissez la taille de segment de mémoire maximale en ajoutant la ligne suivante au fichier:

```
-XmxMaxSizem
```

Où *MaxSize* indique la taille maximale du segment de mémoire, en Mo.

Par exemple, la ligne suivante définit la taille de segment de mémoire maximale à 1GB:

```
-Xmx1024m
```

3. Facultatif : Définissez la taille de segment de mémoire minimale en ajoutant la ligne suivante au fichier:

```
-XmsMinSizem
```

Où *MinSize* indique la taille minimale du segment de mémoire, en Mo. L'augmentation de la taille de segment de mémoire minimale par rapport à la valeur par défaut peut réduire le temps nécessaire au démarrage du serveur mqweb.

Par exemple, la ligne suivante définit la taille de segment de mémoire minimale sur 512MB:

```
-Xms512m
```

4. Facultatif : Définissez la taille du cache de classe en ajoutant la ligne suivante au fichier:

```
-XscmxSizem
```

Où *Taille* indique la taille du cache de classe, en Mo.


Par exemple, la ligne suivante définit la taille du cache de classe sur 100MB:

```
-Xscmx100m
```

Le cache de classes partagées Java est utilisé pour stocker des données telles que les classes chargées et le code compilé AOT (Ahead-Of-Time).

Le cache de classe réduit considérablement le temps nécessaire au démarrage du serveur mqweb. La première fois que le serveur mqweb est démarré, le cache de classe est créé et le démarrage du serveur peut prendre un temps considérable. Les redémarrages ultérieurs du serveur seront beaucoup plus rapides car les classes peuvent être chargées à partir du cache de classes partagées.

L'augmentation de la taille du cache de classe par rapport à la valeur par défaut peut réduire le temps nécessaire au démarrage du serveur mqweb.

 Le cache de classe est recréé lorsque le serveur mqweb est démarré sur un autre système z/OS . Par conséquent, le démarrage du serveur mqweb sur un autre système z/OS dans un sysplex peut prendre beaucoup plus de temps que le redémarrage du serveur sur le même système.

Notez que les modifications apportées à cette valeur ne prennent effet que lorsque le cache de classe est créé. Le cache de classe est créé lorsque le serveur mqweb est démarré pour la première fois ou après que le cache de classe a été détruit à l'aide de l'utilitaire de cache de classe Java .

5. Obligatoire : Vérifiez que le fichier contient les lignes suivantes pour spécifier le codage de fichier utilisé lorsque REST API traite les données et pour les informations de configuration du tableau de bord utilisateur dans IBM MQ Console:

```
-Dfile.encoding=UTF-8
-Ddefault.client.encoding=UTF-8
```

6. Redémarrez le serveur mqweb.

z/OS Sous z/OS, arrêtez et redémarrez la tâche démarrée du serveur mqweb.

Multi Sur toutes les autres plateformes, entrez les commandes suivantes sur la ligne de commande:

```
endmqweb
strmqweb
```

Structure de fichier du composant d'installation IBM MQ Console et REST API

Deux ensembles de structures de répertoire sont associés au composant d'installation IBM MQ Console et REST API . Une structure de répertoire contient des fichiers qui peuvent être édités. L'autre structure de répertoire contient des fichiers qui ne peuvent pas être édités.

Fichiers éditables

Les fichiers modifiables par l'utilisateur sont définis dans le cadre de l'installation initiale du composant d'installation IBM MQ Console et REST API . Comme ces fichiers peuvent être édités, ils ne sont pas modifiés lors de l'application de la maintenance.

L'emplacement des fichiers modifiables par l'utilisateur dépend du système d'exploitation et du produit qui est installé.

- Dans une installation IBM MQ , les fichiers éditables par l'utilisateur se trouvent dans l'un des répertoires suivants:

– **Linux** **AIX** Sous AIX ou Linux: `/var/mqm/web/installations/installationName`

– **Windows** Sous Windows: `MQ_DATA_PATH\web\installations\installationName`, où `MQ_DATA_PATH` est le chemin de données IBM MQ . Il s'agit du chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.

– **z/OS** Sous z/OS: répertoire qui a été spécifié lors de l'exécution du script `crtmqweb` pour créer la définition de serveur mqweb.

- **V 9.4.0** **Linux** Dans une installation IBM MQ Web Server autonome: `MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST`

où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel la variable d'environnement `MQ_OVERRIDE_DATA_PATH` pointe.

Sous ce répertoire de niveau supérieur, les répertoires et fichiers suivants sont présents:




Répertoires et fichiers	Description
<code>angular.persistence/</code>	Répertoire dans lequel la configuration du tableau de bord IBM MQ Console est stockée.
<code>servers/</code>	Répertoire des serveurs WebSphere Liberty .

Répertoires et fichiers	Description
<code>servers/mqweb</code>	Répertoire contenant la structure de répertoire du serveur mqweb.
<code>servers/mqweb/logs</code>	Répertoire contenant les journaux du serveur mqweb.
<code>servers/mqweb/logs/console.log</code>	Journal du statut de base du serveur et des messages d'opération.
<code>servers/mqweb/logs/ffdc</code>	Répertoire de sortie de l'outil de diagnostic de premier niveau (FFDC).
<code>servers/mqweb/logs/messages.log</code>	Journal des messages d'exécution du serveur mqweb, y compris IBM MQ Console et REST API. Les messages plus anciens sont stockés dans des fichiers appelés <code>messages_timestamp.log</code> .
<code>servers/mqweb/logs/trace.log</code>	Journal de trace du serveur mqweb, y compris IBM MQ Console et REST API. L'ancienne trace est stockée dans des fichiers appelés <code>trace_timestamp.log</code> . Ces fichiers existent uniquement si la fonction de trace est activée.
<code>servers/mqweb/logs/state</code>	Etat spécifique au serveur.
<code>servers/mqweb/server.xml</code>	Fichier de configuration du serveur principal. Ce fichier est en lecture seule. Editez le fichier <code>mqwebuser.xml</code> pour remplacer la configuration par défaut.
<code>servers/mqweb/mqwebuser.xml</code>	Fichier de configuration pour IBM MQ Console et REST API. Les paramètres configurés dans ce fichier remplacent la configuration par défaut. Vous devez être un <u>utilisateur privilégié</u> pour pouvoir éditer ce fichier.
<code>servers/mqweb/resources</code>	Répertoire contenant diverses ressources serveur, telles que des magasins de clés.
<code>servers/mqweb/workarea</code>	Répertoire créé par le serveur lors de son fonctionnement. Ce répertoire est créé après la première exécution du serveur.

Fichiers non modifiables

Les fichiers non modifiables sont définis dans le cadre de l'installation initiale du composant d'installation IBM MQ Console et REST API . Ces fichiers sont mis à jour lorsque la maintenance est appliquée.

L'emplacement des fichiers non modifiables dépend du système d'exploitation et du produit installé.

- Dans une installation IBM MQ , les fichiers non modifiables se trouvent dans l'un des répertoires suivants:
 -  Sous AIX, Linux, and Windows: `MQ_INSTALLATION_PATH/web`
 -  Sous IBM i: `MQ_INSTALLATION_PATH/web`
 -  Sous z/OS: `installation_directory/web/`

où *répertoire_installation* est le chemin d'installation de IBM MQ for z/OS UNIX System Services Components .

- **V 9.4.0** **Linux** Dans une installation IBM MQ Web Server autonome, répertoire dans lequel le fichier d'installation IBM MQ Web Server a été décompressé.

La structure de répertoire et les fichiers suivants sont présents à cet emplacement:

Répertoires et fichiers	Description
bin/	Répertoire contenant les commandes WebSphere Liberty . Vous devez être un <u>utilisateur privilégié</u> pour pouvoir exécuter des scripts dans ce répertoire.
mq/	Structure de répertoire contenant diverses ressources IBM MQ .
mq/apps/	Répertoire contenant les applications IBM MQ Console et REST API .
mq/etc/	
mq/etc/mqweb.xml	Fichier de configuration en lecture seule pour le serveur mqweb. Editez le fichier mqwebuser.xml pour apporter des modifications à la configuration.
mq/libs	Répertoire contenant les bibliothèques partagées à utiliser par IBM MQ Console et REST API.
mq/samp	Répertoire contenant des exemples.
mq/samp/configuration	Répertoire contenant des exemples de fichiers de configuration pouvant être copiés dans le fichier mqwebuser.xml .

Sauvegarde et restauration de la configuration de votre serveur mqweb

Vous pouvez sauvegarder votre configuration de serveur mqweb et la restaurer au même emplacement ou à un autre emplacement.

Avant de commencer

Avant de pouvoir restaurer votre configuration de serveur mqweb, vous devez installer IBM MQ, ou le IBM MQ Web Server autonome, sur le système sur lequel vous souhaitez restaurer le serveur mqweb. Dans une installation IBM MQ Web Server autonome, vous devez créer le serveur mqweb en suivant les étapes de la rubrique «[Configuration du IBM MQ Web Server autonome](#)», à la page 887.

Pourquoi et quand exécuter cette tâche


Suivez la procédure de cette tâche pour sauvegarder et restaurer votre configuration de serveur mqweb. Si vous restaurez votre serveur mqweb à un autre emplacement, vous devez mettre à jour la configuration du serveur mqweb pour vous assurer que les références aux fichiers sont correctes.


V 9.4.0 Vous pouvez également utiliser cette procédure pour migrer un serveur mqweb qui s'exécute actuellement dans une installation IBM MQ vers une installation IBM MQ Web Server autonome.


Procédure

1. Pour sauvegarder la configuration du serveur mqweb, copiez tous les fichiers du répertoire contenant la configuration du serveur mqweb dans votre emplacement de sauvegarde.

- Dans une installation IBM MQ , copiez le contenu du répertoire suivant:

–  Sous AIX ou Linux: `/var/mqm/web/installations/installationName`


–  Sous Windows: `MQ_DATA_PATH\web\installations\installationName`, où `MQ_DATA_PATH` est le chemin de données IBM MQ . Il s'agit du chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.


–  Sous z/OS: le répertoire utilisateur WebSphere Liberty qui a été spécifié lors de l'exécution du script `crtmqweb` pour créer la définition de serveur mqweb.


-  Dans une installation IBM MQ Web Server autonome, copiez le contenu du répertoire `MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST` , où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel pointe la variable d'environnement `MQ_OVERRIDE_DATA_PATH` .

2. Pour restaurer la configuration du serveur mqweb, remplacez le contenu du répertoire qui contient la configuration du serveur mqweb par les fichiers que vous avez copiés à l'étape «1», à la page 919.

- Dans une installation IBM MQ , remplacez le contenu du répertoire suivant:

–  Sous AIX ou Linux: `/var/mqm/web/installations/installationName`

–  Sous Windows: `MQ_DATA_PATH\web\installations\installationName`, où `MQ_DATA_PATH` est le chemin de données IBM MQ . Il s'agit du chemin de données sélectionné lors de l'installation de IBM MQ. Par défaut, ce chemin est `C:\ProgramData\IBM\MQ`.


–  Sous z/OS: le répertoire utilisateur WebSphere Liberty qui a été spécifié lors de l'exécution du script `crtmqweb` pour créer la définition de serveur mqweb.

-  Dans une installation IBM MQ Web Server autonome, remplacez le contenu du répertoire `MQ_OVERRIDE_DATA_PATH/web/installations/MQWEBINST` , où `MQ_OVERRIDE_DATA_PATH` est le répertoire de données IBM MQ Web Server vers lequel pointe la variable d'environnement `MQ_OVERRIDE_DATA_PATH` .

3. Définissez la propriété des fichiers que vous avez restaurés à l'étape «2», à la page 919 afin que l'ID utilisateur du serveur mqweb puisse accéder aux fichiers.

4. Si vous avez restauré la configuration du serveur mqweb à un autre emplacement, modifiez la valeur des propriétés de la configuration du serveur mqweb qui font référence aux fichiers du répertoire de configuration du serveur mqweb précédent.

- a) Avant d'exécuter les commandes `setmqweb` ou `dspmqweb` , définissez votre environnement pour qu'il pointe vers votre configuration de serveur mqweb.

-  Sous z/OS, définissez la variable d'environnement `WLP_USER_DIR` de sorte qu'elle pointe vers votre configuration de serveur mqweb, en entrant la commande suivante:

```
export WLP_USER_DIR=WLP_user_directory
```

où `WLP_user_directory` est le nom du répertoire transmis à la commande `crtmqweb` .

Pour plus d'informations, voir [Création du serveur mqweb](#).

-  Dans une installation IBM MQ Web Server autonome, définissez la variable d'environnement `MQ_OVERRIDE_DATA_PATH` sur le répertoire de données IBM MQ Web Server .

- Dans tous les autres environnements, vous n'avez pas besoin d'effectuer d'actions pour définir votre environnement.
- b) Affichez la valeur de toutes les propriétés configurables du serveur mqweb qu'un utilisateur a modifiées. Entrez la commande suivante :

```
dspmqweb properties -u
```

- c) Si la propriété **remoteKeyfile** est affichée, vérifiez la valeur de la propriété.

Si la valeur de la propriété fait référence à un chemin de fichier dans le répertoire de configuration précédent du serveur mqweb, modifiez la valeur pour faire référence au chemin de fichier dans le nouveau répertoire de configuration du serveur mqweb. Exécutez la commande suivante pour modifier la valeur de la propriété **remoteKeyfile** :

```
setmqweb properties -k remoteKeyfile -v path_to_keyfile
```

- d) Affichez la configuration du gestionnaire de files d'attente éloignées du serveur mqweb. Entrez la commande suivante :

```
dspmqweb remote -a
```

- e) Si l'une des propriétés suivantes s'affiche, vérifiez la valeur de la propriété:

- **globalTrustStorePath**
- **globalKeyStorePath**
- **ccdtURL**
- **keyStorePath**
- **trustStorePath**

Modifiez la valeur de toute propriété qui fait référence à un chemin de fichier dans le répertoire de configuration du serveur mqweb précédent pour faire référence au chemin de fichier dans le nouveau répertoire de configuration du serveur mqweb. Exécutez la commande **setmqweb remote** pour modifier la valeur de chaque propriété. Par exemple, pour modifier la valeur de la propriété **keyStorePath** pour le gestionnaire de files d'attente éloignées avec le nom unique **remote-QM1**, exécutez la commande suivante:

```
setmqweb remote -uniqueName remote-QM1 -keyStorePath new_keystore_path
```

Pour plus d'informations, voir [setmqweb remote \(set mqweb server remote queue manager configuration\)](#).

Windows MQ Adv. Linux MQ Adv. VUE MQ Adv. z/OS Définition d'une connexion Aspera gateway sur les plateformes Linux ou Windows

IBM Aspera faspio Gateway fournit un tunnel TCP/IP rapide qui peut augmenter considérablement le débit du réseau pour IBM MQ. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows.





Pourquoi et quand exécuter cette tâche

Aspera gateway peut être utilisé pour améliorer les performances des canaux de gestionnaire de files d'attente. Elle est particulièrement efficace si le réseau a un temps d'attente élevé ou a tendance à perdre des paquets, et elle est généralement utilisée pour accélérer la connexion entre les gestionnaires de files d'attente dans différents centres de données.

Remarque : Pour un réseau rapide qui ne perd pas de paquets, il y a une diminution des performances lors de l'utilisation de Aspera gateway. Il est donc important de vérifier les performances du réseau avant et après la définition d'une connexion Aspera gateway .

Vous définissez un Aspera gateway à chaque extrémité de la connexion réseau IP, puis utilisez TCP/IP pour connecter les canaux du gestionnaire de files d'attente à chaque passerelle. Un gestionnaire de files d'attente n'a pas besoin de s'exécuter sur la même machine que le Aspera gateway qu'il utilise et plusieurs gestionnaires de files d'attente peuvent utiliser la même passerelle.

Pour utiliser le Aspera gateway, vous devez disposer d'une ou de plusieurs des autorisations suivantes:

-  IBM MQ Advanced for Multiplatforms
-  IBM MQ Appliance
-  IBM MQ Advanced for z/OS VUE
-  IBM MQ Advanced for z/OS

Vous pouvez déployer le Aspera gateway sur l'une des plateformes suivantes:

- Linux for x86-64
- Linux on Power Systems - Little Endian
- Linux for IBM Z
- Windows -pour plus d'informations sur la prise en charge des plateformes sous Windows, voir la [documentation IBM Aspera faspio Gateway](#).

Vous ne pouvez utiliser la Aspera gateway que pour les messages IBM MQ, sauf si la passerelle fait l'objet d'une autorisation d'utilisation distincte.

Les gestionnaires de files d'attente qui utilisent Aspera gateway peuvent s'exécuter sur n'importe quelle plateforme prise en charge. Pour la liste complète des plateformes prises en charge, voir [Icônes utilisées dans la documentation du produit](#).


Pour chaque gestionnaire de files d'attente qui ne se trouve pas sur la même machine que le Aspera gateway qu'il utilise, vérifiez que vous disposez d'une connexion réseau rapide entre le gestionnaire de files d'attente et Aspera gateway.

Vous utilisez un fichier `toml` pour créer une définition de passerelle qui définit les ports entrants et sortants utilisés par la passerelle. Un exemple de fichier `toml` est fourni avec Aspera gateway. La définition de passerelle sortante définit la connexion entre le gestionnaire de files d'attente local et la passerelle, et entre la passerelle locale et la passerelle distante. La définition de passerelle entrante définit la connexion entre la passerelle distante et la passerelle locale, et entre la passerelle locale et le gestionnaire de files d'attente local.

Les étapes suivantes fournissent un guide de base pour l'initiation et l'exécution. Pour plus d'informations, voir la [documentation IBM Aspera faspio Gateway](#).









Procédure

1. Obtenez l'image d'installation d' Aspera gateway .

 Pour Multiplatforms, vous téléchargez le Aspera gateway à partir de Passport Advantage. Le téléchargement est intitulé "IBM Aspera faspio Continuous Delivery Release for IBM MQ V9.4 Multiplatform Multilingual eAssembly". Elle est fournie en tant qu'image Continuous Delivery (CD) uniquement en raison du rythme des changements dans cette zone, ce qui signifie que des mises à jour sont nécessaires à la fréquence des éditions CD et que vous pouvez l'installer sur n'importe quel système IBM MQ disposant d'une autorisation d'utilisation IBM MQ Advanced for Multiplatforms ou IBM MQ Appliance . Pour télécharger cet eAssembly, accédez à [Downloading IBM MQ 9.4](#) , puis cliquez sur l'onglet correspondant à l'édition requise. L' eAssembly contient des images d'installation pour toutes les plateformes sur lesquelles la passerelle est disponible. L' eAssembly contient également un fichier `ibm-faspio-license.zip` , qui contient un fichier de licence.

MQ Adv. VUE > MQ Adv. z/OS Si votre système IBM MQ dispose de droits IBM MQ Advanced for z/OS VUE ou IBM MQ Advanced for z/OS, vous obtenez le Aspera gateway du composant Connector Pack qui fait partie de l'installation SMP/E.

MQ Adv. VUE > MQ Adv. z/OS Les fichiers pour IBM MQ Advanced for z/OS VUE et IBM MQ Advanced for z/OS sont les suivants:



Tableau 53. Noms de fichier et numéros de version faspio par plateforme et version IBM MQ		
Plateforme	Nom de fichier	numéro de version faspio
Linux for x86-64	  M0GV DEN . zip	1.3.4
Linux on Power Systems - Little Endian	  M0GV FEN . zip	1.3.4
Linux for IBM Z	  M0GV GEN . zip	1.3.4
Windows	  M0GV HEN . zip	1.3.4

Remarque : Aspera gateway ne peut pas s'exécuter en mode natif sous z/OS.

MQ Adv. VUE > MQ Adv. z/OS Outre les images d'installation, le répertoire fasp contient M05QKEN . zip, qui contient un fichier de licence.

2. Copiez l'image d'installation Aspera gateway sur les deux machines qui exécuteront la passerelle, puis procédez à l'extraction et à l'installation de la passerelle.

Utilisez le fichier de licence contenu dans `ibm-faspio-license.zip` (Multiplatforms) ou `M05QKEN.zip` (z/OS). Pour plus d'informations, voir la documentation IBM Aspera faspio Gateway :

-  [Installation sur Linux](#)
-  [Installation sur Windows](#)

3. Configurez et sécurisez chaque passerelle.



Pour plus d'informations, voir la documentation IBM Aspera faspio Gateway :

- [Configuration du fichier de configuration de la passerelle](#)
- [Sécurisation de la passerelle](#)

4. A chaque extrémité de la connexion réseau, modifiez la définition de canal pour vous connecter au port sur lequel la passerelle locale est à l'écoute.

5. Démarrez chaque service de passerelle.

Pour plus d'informations, voir la documentation IBM Aspera faspio Gateway :

-  [Lancement sur Linux](#)
-  [Lancement sur Windows](#)

6. Redémarrez les canaux.

Vos gestionnaires de files d'attente communiquent désormais via une connexion Aspera gateway .


Exemple

Cet exemple définit une connexion Aspera gateway sur deux machines exécutant Linux. La configuration est la suivante :

- L'adresse IP de la machine de passerelle locale est 9.20.193.107. L'adresse IP de la machine de passerelle distante est 9.20.192.115.
- Le gestionnaire de files d'attente local s'exécute sur une machine dont l'adresse IP est 9.20.121.5. Le gestionnaire de files d'attente éloignées s'exécute sur une machine dont l'adresse IP est 9.20.121.25. Les deux gestionnaires de files d'attente sont en mode écoute sur le port 1414.
- Le canal de gestionnaire de files d'attente sur le gestionnaire de files d'attente local est modifié pour se connecter au Aspera gateway local à l'aide de **conname** 9.20.193.107(1500). Le canal de gestionnaire de files d'attente sur le gestionnaire de files d'attente éloignées est modifié pour se connecter au Aspera gateway éloigné à l'aide de **conname** 9.20.192.115(1500).
- Depuis IBM Aspera faspio Gateway 1.2, TLS est activé par défaut. Si vous souhaitez configurer TLS avec la passerelle, voir [Sécurisation de la passerelle](#) dans la documentation IBM Aspera faspio Gateway .

1. Définissez une connexion Aspera gateway sur la machine de passerelle locale:

- Installez Aspera gateway :

–  Sous Linux, utilisez la commande suivante:

```
rpm -ivh ibm-faspio-gateway-<version>.x86_64.rpm
```

- Modifiez le fichier `gateway.toml` dans le répertoire créé par l'installation:

Editez le fichier pour définir les définitions de passerelle locale.

```
[[bridge]]
  name = "Outbound"
  [bridge.local]
    protocol = "tcp"
    host = "9.20.193.107"
    port = 1500
  tls_enabled = false

  [bridge.forward]
    protocol = "fasp"
    host = "9.20.192.115"
    port = 1600
  tls_enabled = false

[[bridge]]
  name = "Inbound"
  [bridge.local]
    protocol = "fasp"
    host = "9.20.193.107"
    port = 1600
  tls_enabled = false

  [bridge.forward]
    protocol = "tcp"
    host = "9.20.121.5"
    port = 1414
  tls_enabled = false
```

- Copiez le fichier `aspera-license` depuis `ibm-faspio-license.zip` (Multiplatforms) ou `M05QKEN.zip` (z/OS) dans `/usr/local/etc/faspio/`.

2. Répétez l'étape précédente pour définir une connexion Aspera gateway sur la machine de passerelle distante.

- Modifiez le fichier `gateway.toml` dans le répertoire créé par l'installation. Editez le fichier pour définir les définitions de passerelle distante:

```
[[bridge]]
  name = "Outbound"
  [bridge.local]
    protocol = "tcp"
    host = "9.20.193.107"
```

```

        port = 1500
    tls_enabled = false

    [bridge.forward]
        protocol = "fasp"
        host = "9.20.192.115"
        port = 1600
    tls_enabled = false

[[bridge]]
    name = "Inbound"
    [bridge.local]
        protocol = "fasp"
        host = "9.20.193.107"
        port = 1600
    tls_enabled = false

    [bridge.forward]
        protocol = "tcp"
        host = "9.20.121.5"
        port = 1414
    tls_enabled = false

```

- Copiez le fichier `aspera-license` depuis `ibm-faspio-license.zip` (Multiplatforms) ou `M05QKEN.zip` (z/OS) dans `/usr/local/etc/faspio/`.
3. A chaque extrémité de la connexion, modifiez la définition de canal pour qu'elle se connecte au port sur lequel la passerelle locale est à l'écoute.
 - Modifiez le canal du gestionnaire de files d'attente sur le gestionnaire de files d'attente local pour qu'il se connecte au Aspera gateway local à l'aide de **conname** `9.20.193.107(1500)`.
 - Modifiez le canal de gestionnaire de files d'attente sur le gestionnaire de files d'attente éloignées pour qu'il se connecte au Aspera gateway distant à l'aide de **conname** `9.20.192.115(1500)`.
 4. Démarrez la passerelle locale en exécutant la commande suivante sur la machine de la passerelle locale:

•  Linux

```
sudo systemctl start faspio-gateway
```

5. Démarrez la passerelle distante en exécutant la commande suivante sur la machine de la passerelle distante:

•  Linux

```
sudo systemctl start faspio-gateway
```

6. Redémarrez les canaux.

Que faire ensuite

Le Aspera gateway transmet les données qu'il reçoit, sans les interpréter d'aucune manière. Cela signifie que vous pouvez configurer TLS entre les canaux de gestionnaire de files d'attente qui utilisent Aspera gateway car la connexion de passerelle n'est pas consciente de l'établissement de liaison TLS. Cela signifie également que les gestionnaires de files d'attente sur n'importe quelle plateforme IBM MQ prise en charge peuvent utiliser Aspera gateway.

Pour utiliser un gestionnaire de files d'attente multi-instance avec la passerelle, configurez les définitions de passerelle pour chaque instance du gestionnaire de files d'attente.

Remarque : Le Aspera gateway a été testé uniquement avec les canaux de gestionnaire de files d'attente. Elle n'a pas été testée avec des canaux client. En effet, l'utilisation envisagée pour Aspera gateway consiste à connecter des gestionnaires de files d'attente éloignées sur un réseau lent, alors que les applications client se connectent généralement à des gestionnaires de files d'attente dans un centre de données local sur un réseau rapide.

Référence associée

«Type de communication à utiliser», à la page 17

Différentes plateformes prennent en charge différents protocoles de communication. Votre choix de protocole de transmission dépend de votre combinaison de IBM MQ MQI client et de plateformes serveur.

[Documentation IBM Aspera faspio Gateway](#)

Multi Configuration de IBM MQ pour une utilisation avec le service de décompte IBM Cloud Private

Configuration de IBM MQ en vue de son utilisation avec le service de décompte IBM Cloud Private pour générer des rapports et afficher des informations sur le démarrage et l'utilisation du gestionnaire de files d'attente.

Avant de commencer

Avant de configurer vos gestionnaires de files d'attente IBM MQ pour utiliser un service IBM Cloud Private, vous devez disposer d'un compte IBM Cloud. Pour créer votre compte, voir [Inscription à IBM Cloud](#).

Pourquoi et quand exécuter cette tâche

A l'aide du [service de décompte IBM Cloud Private](#), vous pouvez connecter vos produits IBM sur site à votre instance de service dans IBM Cloud Private et voir tous les produits enregistrés dans votre organisation dans un tableau de bord unique.

Vous pouvez configurer et connecter vos gestionnaires de files d'attente AIX, Linux et Windows à votre instance de service de décompte et voir leurs informations de démarrage et d'utilisation. Toutefois, sur les plateformes autres que les environnements Linux Container, les données ne peuvent pas être utilisées pour la prise en charge des licences de tarification horaire basée sur le conteneur.

Pour enregistrer les données d'utilisation d'un type de licence VPC mensuelle, au lieu de la métrique de licence horaire par défaut, définissez la variable d'environnement `AMQ_LICENSING_METRIC=VPCMonthlyPeak`. Ainsi, le gestionnaire de files d'attente télécharge les données relatives aux types de licence VPC mensuels, au lieu du comportement par défaut de téléchargement des données relatives aux licences basées sur un conteneur horaires.

Utilisez les attributs suivants avec la strophe `ReportingService` du fichier `qm.ini` :

APIKeyFile

Emplacement du fichier texte avec la valeur **APIKey** de l'instance de service de décompte.

CapacityReporting

Écrit périodiquement les messages du journal des erreurs dans les journaux AMQERR au format suivant:

```
4/22/2020 01:44:29 PM - Process(1274.1) User(bld-adm) Program(amqmgr0)
Host(8b3b83f2bc7d) Installation(Docker)
VRMF(9.2.0.0)
Time(2020-04-22T13:44:29.295Z)
ArithInsert1(300)
CommentInsert1(8.5)
CommentInsert2(IBM MQ Advanced)
```

Les informations générées par l'attribut **CapacityReporting** sont insérées dans le message AMQ5064, qui vous permet de mieux comprendre à quel point IBM MQ votre entreprise utilise:

AMQ5064

Ce gestionnaire de files d'attente est en cours d'exécution depuis 300 secondes. Il s'exécute actuellement avec des coeurs 8.5. Le type de licence est IBM MQ Advanced.

Gravité

0 : Information

Explication

Il s'agit d'un message d'information pour le suivi de l'utilisation.

Réponse

Néant.

LicensingGroup

Groupe de facturation auquel appartient le gestionnaire de files d'attente. Cela affecte la façon dont les données sont regroupées dans les rapports générés par le service de décompte.

ServiceURL

Adresse du service IBM Cloud Private .

ServiceProxy

URL et port du proxy HTTP pouvant être utilisés si les gestionnaires de files d'attente ne disposent pas d'un accès direct au réseau sur lequel s'exécute le service de décompte.

Vous pouvez voir les hôtes sur lesquels vos produits sont installés, les versions de produit que vous utilisez et les plateformes sur lesquelles elles sont exécutées. A partir des statistiques d'utilisation générales affichées pour chaque produit, vous pouvez avoir une vue globale de la densité des charges de travail. Pour IBM MQ, vous pouvez voir quels gestionnaires de files d'attente sont les plus utilisés et quels gestionnaires de files d'attente ont des charges de travail plus légères.

Lorsqu'un gestionnaire de files d'attente est configuré pour se connecter à une instance du service de décompte, les informations suivantes sont signalées à IBM Cloud Private:

- IBM MQ nom de gestionnaire de files d'attente
- Identificateur du gestionnaire de files d'attente IBM MQ
- Répertoire racine d'installation de IBM MQ
- Composants installés d'IBM MQ (nom et version)
- Nom d'hôte
- Nom du système d'exploitation hôte
- Version du système d'exploitation hôte
- Informations sur l'utilisation du coeur de processeur virtuel (VPC) pour le gestionnaire de files d'attente IBM MQ

Vous pouvez surveiller les métriques d'utilisation VPC de votre gestionnaire de files d'attente dans le tableau de bord de votre instance de service de décompte.

Procédure

- Configurez un gestionnaire de files d'attente à utiliser avec l'instance de service de décompte sur IBM Cloud Private.
- Connectez-vous au service de décompte IBM Cloud Private via un proxy HTTP.
- Identifiez et résolvez les incidents liés à la connexion au service de décompte d' IBM Cloud Private .

Référence associée

[Grille tarifaire des coeurs de processeur virtuels \(CPV\)](#)

Multi Configuration d'un gestionnaire de files d'attente à utiliser avec l'instance de service de décompte sous IBM Cloud Private

Configurez les informations de sécurité et d'enregistrement IBM Cloud pour votre gestionnaire de files d'attente, puis connectez-vous à l'instance de service de décompte que vous avez déjà créée.

Pourquoi et quand exécuter cette tâche

Le tableau de bord de votre instance de [service de décompte IBM Cloud Private](#) affiche des données uniquement pour les gestionnaires de files d'attente configurés pour inclure les informations de sécurité et d'enregistrement IBM Cloud Private .

Procédure

1. Suivez les étapes documentées IBM Cloud Private pour créer un ID de service à l'adresse suivante: [Création d'un ID de service à l'aide de l'interface de ligne de commande IBM Cloud Private](#).
2. Suivez les étapes documentées IBM Cloud Private pour créer une clé d'API à l'adresse suivante: [API de gestion des clés d'API](#).
3. Téléchargez les certificats TLS à partir du cluster IBM Cloud Private .
Notez l'emplacement dans lequel vous avez téléchargé les certificats. Vous pouvez ajouter les certificats téléchargés au référentiel de clés de votre gestionnaire de files d'attente, à l'étape «9», à la page 928.
4. Créez un fichier texte `apikeyfile.txt` et ajoutez la valeur **API key** que vous avez copiée dans la tâche précédente.
Notez l'emplacement du fichier `apikeyfile.txt` pour pouvoir inclure le chemin d'accès à ce fichier à l'étape 8. Ce fichier doit être lisible par l'utilisateur du gestionnaire de files d'attente (`mqm` sur les systèmes AIX and Linux). Le fichier doit contenir uniquement le **API key** lui-même, et non un contenu JSON, par exemple `d9c11b45-4dda-4de4-c0b2-2e4e1004dc64`.
5. Créez le gestionnaire de files d'attente, par exemple `QM1`.
Pour plus d'informations, voir [Création et gestion de gestionnaires de files d'attente sur Multiplatforms](#).
6. Démarrez le gestionnaire de files d'attente `QM1`.
Pour plus d'informations, voir [Démarrage d'un gestionnaire de files d'attente](#).
7. N'oubliez pas de configurer votre environnement de ligne de commande IBM MQ avant d'exécuter des commandes IBM MQ .
Exécutez la commande **setmqenv**.

AIX Sous AIX :

```
./usr/mqm/bin/setmqenv -s
```

Linux Sous Linux :

```
./opt/mqm/bin/setmqenv -s
```

Windows Sous Windows :

```
"C:\Program Files\IBM\MQ\bin\setmqenv.cmd" -n installation name
```

8. Créez un magasin de clés de confiance SSL pour le gestionnaire de files d'attente `QM1`.

AIX Commencez à créer le magasin de clés de confiance sous AIX:

```
runmqakm -keydb -create -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms -expire 30 -stash
```

Linux Sous Linux :

```
runmqakm -keydb -create -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms -expire 30 -stash
```

Windows Sous Windows :

```
runmqakm -keydb -create -db "MQ data directory\qmgrs\QM1\ssl\key.kdb" -pw password -type cms -expire 30 -stash
```

9. Ajoutez les certificats numériques que vous avez téléchargés à l'étape «3», à la page 927 au référentiel de clés du gestionnaire de files d'attente.

AIX Sous AIX :

```
runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms -label RootCA -file Download_location/RootCA.crt -format ascii -trust enable  
runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms -label ServerCert -file Download_location/CERT.crt -format ascii -trust enable
```

Linux Sous Linux :

```
runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms -label RootCA -file Download_location/RootCA.crt -format ascii -trust enable  
runmqakm -cert -add -db MQ data directory/qmgrs/QM1/ssl/key.kdb -pw password -type cms -label ServerCert -file Download_location/CERT.crt -format ascii -trust enable
```

Windows Sous Windows :

```
runmqakm -cert -add -db "MQ data directory\qmgrs\QM1\ssl\key.kdb" -pw password -type cms -label RootCA -file "Download_location\RootCA.crt" -format ascii -trust enable  
runmqakm -cert -add -db "C:\ProgramData\IBM\MQ\qmgrs\QM1\ssl\key.kdb" -pw password -type cms -label ServerCert -file "Download_location\CERT.crt" -format ascii -trust enable
```

10. Ajoutez la nouvelle section ReportingService avec le chemin apikeyfile au fichier qm.ini du gestionnaire de files d'attente:

```
ReportingService:  
APIKeyFile=APIKey file location/apikeyfile.txt
```

11. Ajoutez la valeur **API host** au fichier qm.ini .

La section ReportingService contient maintenant le chemin d'accès à apikeyfile et les valeurs **API host (ServiceURL)**:

```
ReportingService:  
APIKeyFile=APIKey file location/apikeyfile.txt  
ServiceURL=https://productinsights-api.ng.bluemix.net
```

Sauvegardez et quittez le fichier qm.ini .

12. Redémarrez le gestionnaire de files d'attente pour que les modifications soient prises en compte.

Il se peut que vous soyez invité à accorder au processus de gestionnaire de files d'attente **amqzmu10** le droit d'accéder au réseau. L'accès est requis pour permettre au gestionnaire de files d'attente de contacter le service de décompte.

13. Affichez les informations relatives au gestionnaire de files d'attente *QM1* dans votre instance de service de décompte.

Lorsque le statut de génération de rapports est actif, les informations de démarrage et d'utilisation de tous les serveurs d'intégration sur le noeud d'intégration spécifié sont signalées au service de décompte. Les informations d'utilisation sont mises à jour toutes les 15 minutes.

14. Facultatif : Arrêtez un gestionnaire de files d'attente de générer des rapports au service de décompte en supprimant la section `ReportingService` du fichier `qm.ini` du gestionnaire de files d'attente et redémarrez le gestionnaire de files d'attente.
15. Facultatif : Vérifiez les informations de diagnostic dans le fichier journal du gestionnaire de files d'attente si le gestionnaire de files d'attente ne parvient pas à communiquer les informations de démarrage ou d'utilisation au service de décompte.

AIX Sous AIX :

```
/var/mqm/qmgrs/QM1/errors/AMQERR0*.log
```

Linux Sous Linux :

```
/var/mqm/qmgrs/QM1/errors/AMQERR0*.log
```

Windows Sous Windows :

```
C:\ProgramData\IBM\MQ\errors\AMQERR0*.log
```

Résultats

Vous avez créé une instance de service de décompte et configuré votre gestionnaire de files d'attente pour qu'il se connecte à l'instance. Vous pouvez voir les informations relatives à votre gestionnaire de files d'attente dans le tableau de bord de l'instance de service de décompte.

Multi Connexion au service de décompte IBM Cloud Private via un proxy HTTP

Si votre gestionnaire de files d'attente s'exécute sur un système qui ne dispose pas d'un accès direct à votre cluster ICP, vous pouvez utiliser un proxy HTTP fourni par votre organisation pour vous connecter à votre instance de service de décompte dans IBM Cloud Private.

Avant de commencer

Vous avez configuré la sécurité, ajouté **API key** et l'URL de service au fichier `qm.ini` de votre gestionnaire de files d'attente.

Pourquoi et quand exécuter cette tâche

Utilisez cette tâche pour configurer votre gestionnaire de files d'attente afin qu'il se connecte à l'instance de service de décompte dans IBM Cloud Private via un proxy HTTP fourni par votre organisation.

Procédure

- Ajoutez un attribut de proxy de service à la section d'enregistrement IBM Cloud Private de votre fichier `qm.ini`.

Vous pouvez définir l'attribut **ServiceProxy** comme suit:

- URL qui inclut le préfixe `http://` et éventuellement le port. Si vous ne spécifiez pas le port, `1080` est utilisé.

```
ReportingService:  
ServiceProxy=http://myorgproxy.net:1080
```

Remarque : Le paramètre **ServiceProxy** doit être défini sur une URL `http://` valide. Les autres protocoles de proxy, par exemple HTTPS et SOCKS, ne sont pas pris en charge.

- Redémarrez votre gestionnaire de files d'attente avant que les modifications ne soient prises en compte.

Multi Traitement des incidents liés à la connexion au service de décompte

Conseils de traitement des incidents liés aux erreurs que vous pouvez rencontrer lorsque vous connectez votre gestionnaire de files d'attente à une instance de service de décompte.

Le gestionnaire de files d'attente ne peut pas s'enregistrer avec ou télécharger les métriques d'utilisation vers le service de décompte configuré

Vérifiez que le gestionnaire de files d'attente a accès au réseau. La valeur **APIKey** dans le fichier de clé d'API est incorrecte. Assurez-vous que le composant IBM Global Security Kit (GSKit) est installé.

Section `qm.ini` non valide

Une section `qm.ini` non valide a été trouvée. Pour plus d'informations, consultez le journal des erreurs.

Paramètre de proxy de service HTTP non valide

La valeur de l'attribut **ServiceProxy** de la section `ReportingService` du gestionnaire de files d'attente n'est pas configurée correctement. Le gestionnaire de files d'attente ne s'enregistre pas auprès du service. Le paramètre **ServiceProxy** doit être défini sur une URL `http://` valide. Les autres protocoles de proxy, par exemple HTTPS et SOCKS, ne sont pas pris en charge.

z/OS Configuring queue managers on z/OS

Use these instructions to configure queue managers on IBM MQ for z/OS.

Before you begin

Before you configure IBM MQ for z/OS, read:

- [IBM MQ for z/OS concepts](#)
- [Planning your IBM MQ environment on z/OS](#)

About this task

After you have installed IBM MQ, you must carry out a number of tasks before you can make it available to users.

Procedure

- See the following subtopics for information on how to configure queue managers on IBM MQ for z/OS.

Related concepts

[z/OS Sources from which you can issue MQSC and PCF commands on IBM MQ for z/OS](#)

Related tasks

[“Création de gestionnaires de files d'attente sur Multiplatforms” on page 7](#)

Avant de pouvoir utiliser des messages et des files d'attente, vous devez créer et démarrer au moins un gestionnaire de files d'attente et ses objets associés. Un gestionnaire de files d'attente gère les ressources qui lui sont associées, en particulier les files d'attente qu'il possède. Il fournit des services de mise en file d'attente aux applications pour les appels et les commandes MQI (Message queuing Interface) permettant de créer, de modifier, d'afficher et de supprimer des objets IBM MQ .

Securing

[“Configuration de la file d'attente répartie” on page 212](#)

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ , y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.


[“Configuration des connexions entre le client et le serveur” on page 17](#)

Pour configurer les liaisons de communication entre IBM MQ MQI clients et les serveurs, déterminez votre protocole de communication, définissez les connexions aux deux extrémités de la liaison, démarrez un programme d'écoute et définissez des canaux.

 [Administering IBM MQ for z/OS](#)

Planning

Related reference

 [Using the IBM MQ for z/OS utilities](#)

Preparing to customize queue managers on z/OS

Use this topic when customizing your queue managers with details of installable features, national language features, and information about testing, and setting up security.

Preparing for customization

The Program Directory lists the contents of the IBM MQ installation tape, the program and service level information for IBM MQ, and describes how to install IBM MQ for z/OS using System Modification Program Extended (SMP/E). Pour les liens de téléchargement des répertoires de programme, voir [IBM MQ for z/OS Program Directory PDF files](#).

When you have installed IBM MQ, you must carry out a number of tasks before you can make it available to users. See the following sections for a description of these tasks:

- [“Setting up IBM MQ for z/OS” on page 935](#)
- [“Testing a queue manager on z/OS” on page 1001](#)
- [Setting up security on z/OS](#)

If you are migrating from a previous version of IBM MQ for z/OS, you do not need to perform most of the customization tasks. See [Maintaining and migrating](#) for more information about the tasks you must perform.

Installable features of IBM MQ for z/OS

IBM MQ for z/OS comprises the following features:

Base

This is required; it comprises all the main functions, including

- Administration and utilities
- Support for CICS, IMS, and batch type applications using the IBM MQ Application Programming Interface, or C++
- Distributed queuing facility (supporting both TCP/IP and APPC communications)

National language features

These contain error messages and panels in all the supported national languages. Each language has a language letter associated with it. The languages and letters are:

C

Simplified Chinese

E

U.S. English (mixed case)

F

French

K

Japanese

U

U.S. English (uppercase)

You must install the US English (mixed case) option. You can also install one or more other languages. (The installation process for other languages requires US English (mixed case) to be installed, even if you are not going to use US English (mixed case).)

IBM MQ for z/OS UNIX System Services Components

This feature is optional. Select this feature if you want to build and run Java applications that use [Jakarta Messaging 3.0](#) or [Java Message Service 2.0](#) to connect to IBM MQ for z/OS.

See [IBM MQ for z/OS Program Directory PDF files](#) for information on installing IBM MQ for z/OS UNIX System Services Components.

IBM MQ for z/OS UNIX System Services Web Components

This feature is optional.

Select this feature if you want to use the IBM MQ Console, or the REST API.

You must install the IBM MQ for z/OS UNIX System Services Components feature, to install this feature.

IBM MQ for z/OS Managed File Transfer

This feature is optional, and should only be installed if you have entitlement for IBM MQ Advanced for z/OS, IBM MQ for z/OS Value Unit Edition (VUE), or IBM MQ for z/OS Managed File Transfer.

Select this feature if you want to use the Managed File Transfer capabilities of IBM MQ for z/OS.

You must install the IBM MQ for z/OS UNIX System Services Components feature, to install this feature.

Libraries that exist after installation

IBM MQ is supplied with a number of separate load libraries. [Table 54 on page 932](#) shows the libraries that might exist after you have installed IBM MQ.

<i>Table 54. IBM MQ libraries that exist after installation</i>	
Name	Description
thlqual.SCSQANLC	Contains the load modules for the Simplified Chinese version of IBM MQ.
thlqual.SCSQANLE	Contains the load modules for the U.S. English (mixed case) version of IBM MQ.
thlqual.SCSQANLF	Contains the load modules for the French version of IBM MQ.
thlqual.SCSQANLK	Contains the load modules for the Japanese version of IBM MQ.
thlqual.SCSQANLU	Contains the load modules for the U.S. English (uppercase) version of IBM MQ.
thlqual.SCSQASMS	Contains source for assembler sample programs.
thlqual.SCSQAUTH	The main repository for all IBM MQ product load modules; it also contains the default parameter module, CSQZPARM. This library must be APF-authorized and in PDS-E format.
thlqual.SCSQCICS	Contains extra load modules that must be included in the CICS DFHRPL concatenation. This library must be APF-authorized and in PDS-E format.
thlqual.SCSQCLST	Contains CLISTS used by the sample programs.
thlqual.SCSQCOBC	Contains COBOL copybooks, including copybooks required for the sample programs.
thlqual.SCSQCOBS	Contains source for COBOL sample programs.
thlqual.SCSQCPPS	Contains source for C++ sample programs.

Table 54. IBM MQ libraries that exist after installation (continued)

Name	Description
thlqual.SCSQC37S	Contains source for C sample programs.
thlqual.SCSQC370	Contains C headers, including headers required for the sample programs.
thlqual.SCSQDEFS	Contains side definitions for C++ and the Db2 DBRMs for shared queuing.
thlqual.SCSQEXEC	Contains REXX executable files to be included in the SYSEXEC or SYSPROC concatenation if you are using the IBM MQ operations and control panels.
thlqual.SCSQFCMD	Contains templates for jobs to create and run Managed File Transfer tasks.
thlqual.SCSQHPPS	Contains header files for C++.
thlqual.SCSQINST	Contains JCL for installation jobs.
thlqual.SCSQLINK	Early code library. Contains the load modules that are loaded at system initial program load (IPL). The library must be APF-authorized.
thlqual.SCSQLOAD	Load library. Contains load modules for non-APF code, user exits, utilities, samples, installation verification programs, and adapter stubs. The library does not need to be APF-authorized and does not need to be in the link list. This library must be in PDS-E format.
thlqual.SCSQMACS	Contains Assembler macros including: sample macros, product macros, and system parameter macros.
thlqual.SCSQMAPS	Contains CICS mapsets used by sample programs.
thlqual.SCSQMSGC	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the Simplified Chinese language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGE	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the U.S. English (mixed case) language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGF	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the French language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGK	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the Japanese language feature for the IBM MQ operations and control panels.
thlqual.SCSQMSGU	Contains ISPF messages to be included in the ISPMLIB concatenation if you are using the U.S. English (uppercase) language feature for the IBM MQ operations and control panels.
thlqual.SCSQMVR1	Contains the load modules for distributed queuing. This library must be APF-authorized and in PDS-E format.
thlqual.SCSQPLIC	Contains PL/I include files.
thlqual.SCSQPLIS	Contains source for PL/I sample programs.

Table 54. IBM MQ libraries that exist after installation (continued)

Name	Description
thlqual.SCSQPnLA	Contains IPCS panels, for the dump formatter, to be included in the ISPLIB concatenation. Also contains panels for IBM MQ sample programs.
thlqual.SCSQPnLC	Contains ISPF panels to be included in the ISPLIB concatenation if you are using the Simplified Chinese language feature for the IBM MQ operations and control panels.
thlqual.SCSQPnLE	Contains ISPF panels to be included in the ISPLIB concatenation if you are using the U.S. English (mixed case) language feature for the IBM MQ operations and control panels.
thlqual.SCSQPnLF	Contains ISPF panels to be included in the ISPLIB concatenation if you are using the French language feature for the IBM MQ operations and control panels.
thlqual.SCSQPnLK	Contains ISPF panels to be included in the ISPLIB concatenation if you are using the Japanese language feature for the IBM MQ operations and control panels.
thlqual.SCSQPnLU	Contains ISPF panels to be included in the ISPLIB concatenation if you are using the U.S. English (uppercase) language feature for the IBM MQ operations and control panels.
thlqual.SCSQPROC	Contains sample JCL and default system initialization data sets.
thlqual.SCSQSNLC	Contains the load modules for the Simplified Chinese versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLE	Contains the load modules for the U.S. English (mixed case) versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLF	Contains the load modules for the French versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLK	Contains the load modules for the Japanese versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQSNLU	Contains the load modules for the U.S. English (uppercase) versions of the IBM MQ modules that are required for special purpose function (for example the early code).
thlqual.SCSQTbLC	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the Simplified Chinese language feature for the IBM MQ operations and control panels.
thlqual.SCSQTbLE	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the U.S. English (mixed case) language feature for the IBM MQ operations and control panels.
thlqual.SCSQTbLF	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the French language feature for the IBM MQ operations and control panels.

Table 54. IBM MQ libraries that exist after installation (continued)

Name	Description
thlqual.SCSQTBLK	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the Japanese language feature for the IBM MQ operations and control panels.
thlqual.SCSQTBLU	Contains ISPF tables to be included in the ISPTLIB concatenation if you are using the U.S. English (uppercase) language feature for the IBM MQ operations and control panels.

Note: Do not modify or customize any of these libraries. If you want to make changes, copy the libraries and make your changes to the copies.

Related concepts

[IBM MQ for z/OS concepts](#)

[“Using IBM MQ with IMS” on page 1039](#)

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

[“Using IBM MQ with CICS” on page 1047](#)

To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

[“Using OTMA exits in IMS” on page 1049](#)

Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

Related tasks

[“Setting up communications with other queue managers on z/OS” on page 1009](#)

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

[Administering IBM MQ for z/OS](#)

Related reference

[“Upgrading and applying service to Language Environment or z/OS Callable Services” on page 1047](#)

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS

Setting up IBM MQ for z/OS

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

The best way to configure a queue manager is to carry out the following steps in the order shown:

1. Configure the base queue manager.
2. Configure the channel initiator, which performs queue manager to queue manager communications, and remote client application communication.
3. If you want to encrypt or protect messages, configure Advanced Message Security for z/OS.
4. If you want to use IBM MQ to transfer files, configure Managed File Transfer for z/OS.
5. If you want to use the administrative or messaging REST API, or the IBM MQ Console to manage IBM MQ from a web browser, configure the mqweb server.

This topic leads you through the various stages of setting up IBM MQ after you have successfully installed it. The installation process is described in the Program Directory. Pour les liens de téléchargement des répertoires de programme, voir [IBM MQ for z/OS Program Directory PDF files](#).

Samples are supplied with IBM MQ to help you with your customization. The sample data set members have names beginning with the four characters CSQ4 and are in the library thlqual.SCSQPROC.

Before you perform the customization tasks described in this topic, there are a number of configuration options that you must consider because they affect the performance and resource requirements of IBM MQ for z/OS. For example, you must decide which globalization libraries you want to use.

If you want to automate some of the customization steps, see [“Using IBM z/OSMF to automate IBM MQ” on page 1053](#).

Configuration options

For more information about these options, see [Planification sur z/OS](#).

The description of each task in this section indicates whether:

- The task is part of the process of setting up IBM MQ. That is, you perform the task once when you customize IBM MQ on the z/OS system. (In a parallel sysplex, you must perform the task for each z/OS system in the sysplex, and ensure that each z/OS system is set up identically.)
- The task is part of adding a queue manager. That is, you perform the task once for each queue manager when you add that queue manager.

None of the tasks require you to perform an IPL of your z/OS system, if you use commands to change the various z/OS system parameters, and perform [“Update SYS1.PARMLIB members” on page 950](#) as suggested.

To simplify operations and to aid with problem determination, ensure that all z/OS systems in a sysplex are set up identically, so that queue managers can be quickly created on any system in an emergency.

For ease of maintenance, consider defining aliases to refer to your IBM MQ libraries; for more information, see [Using an alias to refer to an IBM MQ library](#).

Related concepts

[IBM MQ for z/OS concepts](#)

[“Using IBM MQ with IMS” on page 1039](#)

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

[“Using IBM MQ with CICS” on page 1047](#)

To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

[“Using OTMA exits in IMS” on page 1049](#)

Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

Related tasks

[“Setting up communications with other queue managers on z/OS” on page 1009](#)

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

[Administering IBM MQ for z/OS](#)

Related reference

[“Upgrading and applying service to Language Environment or z/OS Callable Services” on page 1047](#)

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

Configuring the z/OS system for IBM MQ

Use these topics as a step by step guide for customizing your IBM MQ for z/OS system.

Identify the z/OS system parameters

Some of the tasks involve updating the z/OS system parameters. You need to know which ones were specified when the system IPL was performed.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*

- You might need to perform this task when migrating from a previous version.

SYS1.PARMLIB(IEASYSpp) contains a list of parameters that point to other members of SYS1.PARMLIB (where pp represents the z/OS system parameter list that was used to perform an IPL of the system).

The entries you need to find are:

For “La fonction APF autorise les bibliothèques de chargement IBM MQ” on page 937:

PROG=xx or APF=aa point to the Authorized Program Facility (APF) authorized library list (member PROGxx or IEFAPFaa)

For “Mise à jour de la liste de liens z/OS et LPA” on page 938:

LNK=kk points to the link list (member LNKLSTkk) LPA=mm points to the LPA list (member LPALSTmm)

For “Update the z/OS program properties table” on page 942:

SCH=xx points to the Program Properties Table (PPT) (member SCHEDxx)

For “Define the IBM MQ subsystem to z/OS” on page 943:

SSN=ss points to the defined subsystem list (member IEFSSNss)

z/OS La fonction APF autorise les bibliothèques de chargement IBM MQ

Attribuez des droits APF à diverses bibliothèques. Certains modules de chargement sont peut-être déjà autorisés.

Remarques :

- Vous devez effectuer cette tâche une fois pour chaque z/OS système sur lequel vous souhaitez exécuter IBM MQ .
- Si vous utilisez des groupes de partage de file d'attente, vous devez vous assurer que les paramètres de IBM MQ sont identiques sur chaque z/OS système dans le sysplex.
- Vous devrez peut-être effectuer cette tâche lors de la migration à partir d'une version précédente.
- Utilisation de Library Look Side (LLA) :
 - Quelques IBM MQ l'utilisation peut entraîner des entrées/sorties (IO) élevées pour charger des modules à partir de bibliothèques. Ces E/S peuvent être réduites en utilisant la fonction LLA du système d'exploitation.
 - Cette IO élevée peut se produire pendant :
 - Applications avec un taux MQCONN/MQDISC élevé, par exemple dans une procédure stockée WLM.
 - Le canal de chargement se termine. Si vous avez des chaînes qui démarrent et s'arrêtent fréquemment et utilisez les sorties de chaîne.
 - Le membre CSVLLAxx dans SYS1.PARMLIB spécifie la configuration LLA. L'inclusion d'un nom de bibliothèque dans l'instruction LIBRARIES signifie qu'une copie du programme sera toujours extraite de VLF (Virtual Lookaside Facility) et ne nécessitera donc généralement pas d'E/S en cas d'utilisation intensive.

L'inclusion dans l'instruction FREEZE signifie qu'il n'y a pas d'E/S pour obtenir les répertoires de concaténation des instructions DD pertinents (cela peut souvent représenter plus d'E/S que la charge du programme lui-même).

Utilisez la commande du système d'exploitation " F LLA , REFRESH " après toute modification apportée à l'une de ces bibliothèques.

Les IBM MQ bibliothèques de chargement thlqual.SCSQAUTH et thlqual.SCSQLINK doivent être autorisées par APF. Vous devez également autoriser par APF les bibliothèques pour votre fonction de langue nationale (thlqual.SCSQANLx et thlqual.SCSQSNLx) et pour la fonction de mise en file d'attente répartie (thlqual.SCSQMVR1).

Cependant, tous les modules de chargement de la zone permanente de programme sont automatiquement autorisés par APF. Il en est de même pour tous les membres de la liste de liens si SYS1.PARMLIB IEASYSpp de PARMLIB contient l'instruction suivante:

```
LNKAUTH=LNKLST
```

LNKAUTH=LNKLST est la valeur par défaut si LNKAUTH n'est pas indiqué.

En fonction de ce que vous choisissez de placer dans la zone permanente de programme ou dans la liste de liens (voir «[Mise à jour de la liste de liens z/OS et LPA](#)», à la page 938), il se peut que vous n'ayez pas besoin de placer les bibliothèques dans la liste de liens APF

Remarque : Vous devez autoriser par APF toutes les bibliothèques que vous incluez dans IBM MQ STEPLIB. Si vous placez une bibliothèque qui n'est pas autorisée par APF dans STEPLIB, la concaténation de l'ensemble de la bibliothèque perd son autorisation APF.

Les listes APF se trouvent dans SYS1.PARMLIB membre PROGxx ou IEAAPFaa. Les listes contiennent les noms des bibliothèques z/OS autorisées par APF. L'ordre des entrées dans les listes n'est pas significatif. Pour plus d'informations sur les listes APF, voir [Liste des bibliothèques autorisées par APF](#) .

Pour plus d'informations sur l'optimisation de votre système, voir [SupportPac MP16](#)

Si vous utilisez des membres PROGxx avec un format dynamique, vous devez uniquement exécuter la commande z/OS SETPROG APF,ADD,DSNAME=h1q.SCSQ XXXX,VOLUME= YYYYYY pour que les modifications soient prises en compte: où XXXX varie en fonction du nom de la bibliothèque et où YYYYYY correspond au volume. Sinon, si vous utilisez un format statique ou des membres IEAAPFaa, vous devez effectuer un IPL sur votre système.

Notez que vous devez utiliser le nom réel de la bibliothèque dans la liste APF. Si vous tentez d'utiliser l'alias de fichier de la bibliothèque, l'autorisation échoue.

Concepts associés

«[Mise à jour de la liste de liens z/OS et LPA](#)», à la page 938

Mettez à jour les bibliothèques LPA avec la nouvelle version des bibliothèques à code initial. Un autre code peut figurer dans la liste de liens ou dans la zone permanente de programme (LPA).

«[Preparing to customize queue managers on z/OS](#)», à la page 931

Use this topic when customizing your queue managers with details of installable features, national language features, and information about testing, and setting up security.

Mise à jour de la liste de liens z/OS et LPA

Mettez à jour les bibliothèques LPA avec la nouvelle version des bibliothèques à code initial. Un autre code peut figurer dans la liste de liens ou dans la zone permanente de programme (LPA).

- Vous devez effectuer cette tâche une fois pour chaque système z/OS sur lequel vous souhaitez exécuter IBM MQ.
- Si vous utilisez des groupes de partage de files d'attente, vous devez actualiser le code initial de chaque gestionnaire de files d'attente du groupe de partage de files d'attente au niveau IBM MQ 9.4.0 avant de migrer l'un des gestionnaires de files d'attente vers IBM MQ 9.4.0.

Installez le code initial le plus récent sur chaque partition logique, puis actualisez les gestionnaires de files d'attente un par un avant la migration. Il n'est pas nécessaire de migrer tous les gestionnaires de files d'attente en même temps.

- Vous devrez peut-être effectuer cette tâche lors de la migration à partir d'une version précédente. Pour plus de détails, voir le répertoire de programme. Pour les liens de téléchargement des répertoires de programme, voir [IBM MQ for z/OS Program Directory PDF files](#).

Remarque : Le fichier pour LPA est spécifique à la version. Si vous utilisez une zone permanente de programme existante dans le système, contactez votre administrateur système pour déterminer la zone permanente de programme à utiliser.

Code initial

Certains modules de chargement IBM MQ doivent être ajoutés à MVS pour que IBM MQ puisse agir en tant que sous-système. Ces modules sont connus sous le nom de code Early et peuvent être exécutés même si un gestionnaire de files d'attente n'est pas actif. Par exemple, lorsqu'une commande d'opérateur est émise sur la console avec un préfixe de commande IBM MQ, ce code Early prend le contrôle et vérifie s'il a besoin de démarrer un gestionnaire de files d'attente ou de transmettre la demande à un gestionnaire de files d'attente en cours d'exécution. Ce code est chargé dans la zone permanente de programme (LPA). Il existe un ensemble de modules précoces, qui sont utilisés pour tous les gestionnaires de files d'attente et qui doivent être au niveau le plus élevé de IBM MQ. Le code initial d'une version ultérieure d' IBM MQ fonctionne avec un gestionnaire de files d'attente avec une version antérieure d' IBM MQ, mais pas le contraire.

Le code initial comprend les modules de chargement suivants:

- CSQ3INI et CSQ3EPX dans la bibliothèque thqual.SCSQLINK
- CSQ3ECMX dans la bibliothèque thqual.SCSQSNL x, où x est votre lettre de langue:
 - thlqual.SCSQSNLE, pour la casse mixte en anglais américain
 - thlqual.SCSQSNLU, pour les majuscules en anglais américain
 - thlqual.SCSQSNLK, pour le japonais
 - thlqual.SCSQSNLF, pour le français
 - thlqual.SCSQSNLC, pour le chinois

IBM MQ inclut une modification utilisateur qui déplace le contenu de la bibliothèque thqual.SCSQSNL i dans thqual.SCSQLINK et en informe SMP/E. Cette modification utilisateur est appelée CSQ8UERL et est décrite dans le document *Program Directory for IBM MQ for z/OS*, pour Long Term Support ou Continuous Delivery. Pour les liens de téléchargement des répertoires de programme, voir [IBM MQ for z/OS Program Directory PDF files](#).

Une fois que vous avez mis à jour le code initial dans les bibliothèques LPA, il est disponible à partir du prochain démarrage du système z/OS (avec l'option CLPA) pour tous les sous-systèmes de gestionnaire de files d'attente ajoutés lors de l'IPL à partir des définitions des membres IEFSSNss dans SYS1.PARMLIB.

Vous pouvez le rendre disponible immédiatement sans IPL pour tout nouveau sous-système de gestionnaire de files d'attente ajouté ultérieurement (comme décrit dans [«Define the IBM MQ subsystem to z/OS»](#), à la page 943) en l'ajoutant à la zone permanente de programme (LPA) comme suit:

- Si vous n'avez pas utilisé CSQ8UERL, exécutez les commandes z/OS suivantes:

```
SETPROG LPA,ADD,MODNAME=(CSQ3INI,CSQ3EPX),DSNAME=thqua1.SCSQLINK
SETPROG LPA,ADD,MODNAME=(CSQ3ECMX),DSNAME=thqua1.SCSQSNL x
```

- Si vous avez utilisé CSQ8UERL, vous pouvez charger le code initial dans la zone permanente de programme à l'aide de la commande z/OS suivante:

```
SETPROG LPA,ADD,MASK=*,DSNAME=thqua1.SCSQLINK
```

- Si vous utilisez Advanced Message Security, vous devez également exécuter la commande z/OS suivante pour inclure un module supplémentaire dans la zone permanente de programme:

```
SETPROG LPA,ADD,MODNAME=(CSQ0DRTM),DSNAME=thqua1.SCSQLINK
```

Si vous avez appliqué la maintenance ou que vous avez l'intention de redémarrer un gestionnaire de files d'attente avec une version ou une édition ultérieure d' IBM MQ, le code initial peut être mis à la disposition des gestionnaires de files d'attente existants en procédant comme suit. Les gestionnaires de files d'attente pour lesquels vous n'effectuez pas ces étapes continuent d'utiliser la version du code initial qu'ils utilisent déjà. Il n'est pas nécessaire d'effectuer ces étapes pour tous les gestionnaires de files

d'attente d'une partition logique, sauf si vous essayez spécifiquement d'appliquer la maintenance à tous les gestionnaires de files d'attente ou de les mettre à jour vers une version ou une édition plus récente d'IBM MQ.

1. Ajoutez-le à la zone permanente de programme (LPA) à l'aide des commandes z/OS SETPROG, comme décrit précédemment dans cette rubrique.
2. Arrêtez le gestionnaire de files d'attente à l'aide de la commande IBM MQ STOP QMGR.
3. Vérifiez que qmgr.REFRESH.QMGR est configuré. Voir [Commandes MQSC, profils et leurs niveaux d'accès](#).
4. Actualisez le code initial du gestionnaire de files d'attente à l'aide de la commande IBM MQ REFRESH QMGR TYPE (EARLY).
5. Redémarrez le gestionnaire de files d'attente à l'aide de la commande IBM MQ START QMGR.

Les commandes IBM MQ STOP QMGR, REFRESH QMGR et START QMGR sont décrites dans les [commandes MQSC](#).

Autre code

Tous les modules de chargement fournis par IBM MQ dans les bibliothèques suivantes sont réentrants et peuvent être placés dans la zone permanente de programme (LPA):


- SCSQAUTH
- SCSQANL x, où x est votre lettre de langue
- SCSQMVR1

Important : Toutefois, si vous placez les bibliothèques dans la zone permanente de programme, chaque fois que vous appliquez la maintenance, vous devez copier manuellement les modules modifiés dans la zone permanente de programme. Par conséquent, il est préférable de placer les bibliothèques de chargement IBM MQ dans la liste de liens, qui peut être mise à jour après la maintenance en exécutant la commande z/OS MODIFY LLA REFRESH.

Pour plus d'informations, voir [Modification du contenu des ensembles de données LNKLST](#) et [Utilisation de la fonction LNKLST dynamique en toute sécurité et correctement](#).



Cela est particulièrement recommandé pour SCSQAUTH afin que vous n'ayez pas à l'inclure dans plusieurs STEPLIBs. Une seule bibliothèque de langues, SCSQANL x, doit être placée dans la zone permanente de programme ou dans la liste de liens. Les bibliothèques de liste de liens sont indiquées dans un membre LNKLSTkk de SYS1.PARMLIB.

La fonction de mise en file d'attente répartie et CICS bridge (mais pas le gestionnaire de files d'attente lui-même) doivent accéder à la bibliothèque d'exécution LE (Language Environment) SCEERUN. Si vous utilisez l'une de ces fonctions, vous devez inclure SCEERUN dans la liste de liens.

 Certains modules sont chargés au démarrage du gestionnaire de files d'attente dans ECSA. Dans les environnements avec contraintes ECSA, vous pouvez placer ces modules dans la zone permanente de programme à la place. Voir [«Placing IBM MQ global modules into the LPA»](#), à la page 940.

Concepts associés

«Update the z/OS program properties table», à la page 942
Some additional PPT entries are needed for the IBM MQ queue manager.

  *Placing IBM MQ global modules into the LPA*

When an IBM MQ for z/OS queue manager starts up, it loads some of its load modules (global modules) into the extended common service area (ECSA). At queue manager shut down the ECSA is freed.

There are 19 global modules, which at IBM MQ 9.3, consumed approximately 1.2 MB of ECSA for each running queue manager.

Note: Although CSQ7GPLM is a global module, it should not be added to the LPA.

In environments that run multiple queue managers for each LPAR, and require a reduction in ECSA consumption due to ECSA or high private constraints, it is possible to place the global modules into the LPA. Placing the global modules of IBM MQ into the LPA is a manual process that requires care, so you should only carry out this procedure if there is a significant need to address ECSA or high private constraints.

If the queue manager cannot find a global module in its STEPLIB, and detects the module is in the LPA, it uses the LPA copy directly, instead of loading a copy of the module into ECSA. Alternatively if the queue managers code is normally loaded from the link list then any global modules in the LPA are loaded in preference to any global modules in the link list.

The z/OS common storage tracking function (see [Using the common storage tracking function](#)) tracks the storage under the each queue manager's MSTR address space for each queue manager and can be used to detect how much space is being use by the global modules.

By default, the global modules are in the SCSQAUTH load library. If the MSTR address space of a queue manager locates SCSQAUTH through the STEPLIB concatenation, the global modules from there are used in preference to any in the LPA and are loaded into ECSA.

The global modules are:

CSQ0GPLM, CSQ3AMGP, CSQ3SSGP, CSQ9PREP,
CSQ9SCNB, CSQGGPLM, CSQMCGLM, CSQMGPLM, CSQRGLM1,
CSQSLD1, CSQVGEPL, CSQVSRX, CSQWDL2, CSQWDL3,
CSQWVZSA, CSQWZDGO, CSQWVZPS, CSQWVGTM, CSQZTDDM

Important:

- The name of the global modules for IBM MQ remain constant across different IBM MQ versions. Therefore, if you load global modules into the LPA, they should be from a single IBM MQ version, and should only be used by queue managers running at the same IBM MQ version.
- If multiple versions of IBM MQ are run on the same LPAR then only one of those can have its global modules in the LPA at any given time.
- If maintenance is applied to an IBM MQ installation which has global modules loaded into the LPA, and that maintenance updates any of the global modules, you should perform the procedure described in the following text again.

Procedure

To put the global modules from a version of IBM MQ into the LPA, perform the following steps:

1. Create a copy of the th1qua1 . SCSQAUTH load library, and its contents, for example: th1qua1 . LOCAL . SCSQAUTH. Ensure that this load library is protected from unauthorized access using your external security manager (ESM).
2. APF authorize the th1qua1 . LOCAL . SCSQAUTH load library; see [“La fonction APF autorise les bibliothèques de chargement IBM MQ”](#) on page 937.
3. Create a new th1qua1 . GLOBAL . SCSQAUTH load library with the same attributes as th1qua1 . LOCAL . SCSQAUTH.

Note: This load library does not need to be APF authorized. Ensure that this load library is protected from unauthorized access using your ESM.

4. Copy the 19 global modules from th1qua1 . LOCAL . SCSQAUTH into th1qua1 . GLOBAL . SCSQAUTH.
5. Delete the 19 global modules from th1qua1 . LOCAL . SCSQAUTH.
6. Place the 19 global modules from th1qua1 . GLOBAL . SCSQAUTH into the LPA, by either:
 - a. Adding th1qua1 . GLOBAL . SCSQAUTH into an LPALSTxx member of SYS1.PARMLIB. You must then IPL the system with the CLPA option to ensure that the library contents are loaded into the PLPA.
 - b. Dynamically adding the modules to the LPA using the following command:

```
SETPROG
LPA,ADD,MODNAME=(CSQ0GPLM,CSQ3AMGP,CSQ3SSGP,CSQ9PREP,CSQ9SCNB,CSQGGPLM,
CSQMCGLM,CSQMGPLM,CSQRGLM1,CSQSLD1,CSQVGEPL,CSQVSRX,CSQWDL2,CSQWDL3,
CSQWVZSA,CSQWZDGO,CSQWVZPS,CSQWVGTM,CSQZTDDM),DSNAME= thlqual.GLOBAL.SCSQAUTH
```

Note: LPA_{STxx} is the preferred long term means of placing modules in LPA.

7. Validate that the modules are in the LPA by issuing the following command:

```
D PROG,LPA,MODNAME=CSQMCGLM
```

The output of the command should indicate the entry and load points of the module if it was successfully loaded into the LPA.

For each queue manager that needs to use the global modules from the LPA, then if you normally place:

1. thlqual.SCSQAUTH in the link list, just stop and start your queue manager. The global modules are loaded from the LPA, and the local modules from the link list.
2. thlqual.SCSQAUTH in the MSTR JCL STEPLIB, change the JCL so that the STEPLIB uses thlqual.LOCAL.SCSQAUTH instead of thlqual.SCSQAUTH. Stop and start the queue manager; the global modules are loaded from the LPA, and the local modules from the STEPLIB.

The CHIN and AMSM JCL can continue to use thlqual.SCSQAUTH as can any IBM MQ applications.

To revert the queue manager to loading the global modules into ECSA perform the following steps:

1. Stop the queue managers
2. Remove the global modules from the LPA, either at the next IPL by removing the LPA_{STxx} definitions or by using the following command:

```
SETPROG LPA,DELETE,MODNAME=(xxx) FORCE=YES
```

3. If thlqual.LOCAL.SCSQAUTH is in the STEPLIB of the queue manager replace it with thlqual.SCSQAUTH.
4. Restart the queue managers.

Related concepts

[“Mise à jour de la liste de liens z/OS et LPA” on page 938](#)

Mettez à jour les bibliothèques LPA avec la nouvelle version des bibliothèques à code initial. Un autre code peut figurer dans la liste de liens ou dans la zone permanente de programme (LPA).

Update the z/OS program properties table

Some additional PPT entries are needed for the IBM MQ queue manager.

- *You must perform this task once for each z/OS system where you want to run IBM MQ.*
- *If you are using queue sharing groups, you must ensure that the settings for IBM MQ are identical on each z/OS system in the sysplex.*
- *You do not need to perform this task when migrating from a previous version.*
- *You do need to perform the CSQ0DSRV part of this task when you require Advanced Message Security.*

A sample containing all the required PPT entries is provided in thlqual.SCSQPROC(CSQ4SCHD). Ensure that the required entries are added to the PPT, which you can find in SYS1.PARMLIB(SCHEDxx).

In z/OS, CSQYASCP is already defined to the operating system with the attributes detailed and no longer needs to be included in a SCHEDxx member of PARMLIB.

The IBM MQ queue manager controls swapping itself. However, if you have a heavily-loaded IBM MQ network and response time is critical, it might be advantageous to make the IBM MQ channel initiator nonswappable, by adding the CSQXJST PPT entry, at the risk of affecting the performance of the rest of your z/OS system.

If you require Advanced Message Security, add the CSQ0DSRV PPT entry.

Issue the z/OS command **SET SCH=xx**, where xx is the suffix of the SCHEDxx member of PARMLIB, for these changes to take effect.

Related concepts

[“Define the IBM MQ subsystem to z/OS” on page 943](#)

Update the subsystem name table and decide on a convention for command prefix strings.

Configuring the queue manager and channel initiator

Use these topics as a step by step guide for configuring the queue manager and channel initiator.

Define the IBM MQ subsystem to z/OS

Update the subsystem name table and decide on a convention for command prefix strings.

Repeat this task for each IBM MQ queue manager. You do not need to perform this task when migrating from a previous version.

Related concepts

[“Create procedures for the IBM MQ queue manager” on page 946](#)

Each IBM MQ subsystem needs a cataloged procedure to start the queue manager. You can create your own or use the IBM-supplied procedure library.

Updating the subsystem name table

When defining the IBM MQ subsystem you must add an entry to the subsystem name table.

The subsystem name table of z/OS, which is taken initially from the SYS1.PARMLIB member IEFSSNss, contains the definitions of formally defined z/OS subsystems. To define each IBM MQ subsystem, you must add an entry to this table, either by changing the IEFSSNss member of SYS1.PARMLIB, or, preferably, by using the z/OS command SETSSI.

IBM MQ subsystem initialization supports parallel processing, so IBM MQ subsystem definition statements can be added both above and below the BEGINPARALLEL keyword in the IEFSSNss table available at z/OS V1.12 and later.

If you use the SETSSI command, the change takes effect immediately, and there is no need to perform an IPL of your system. Ensure you update SYS1.PARMLIB as well, as described in [“Update SYS1.PARMLIB members” on page 950](#) so that the changes remain in effect after subsequent IPLs.

The SETSSI command to dynamically define an IBM MQ subsystem is:

```
SETSSI ADD,S=ssid,I=CSQ3INI,P='CSQ3EPX,cpf,scope'
```

The corresponding information in IEFSSNss can be specified in one of two ways:

- The keyword parameter form of the IBM MQ subsystem definition in IEFSSNss. This is the recommended method.

```
SUBSYS SUBNAME(ssid) INITRTN(CSQ3INI) INITPARM('CSQ3EPX,cpf,scope')
```

- The positional parameter form of the IBM MQ subsystem definition.

```
ssid,CSQ3INI,'CSQ3EPX,cpf,scope'
```

Do not mix the two forms in one IEFSSNss member. If different forms are required, use a separate IEFSSNss member for each type, adding the SSN operand of the new member to the IEASYSpp SYS1.PARMLIB member. To specify more than one SSN, use SSN=(aa,bb,...) in IEASYSpp.

In the examples,

ssid

The subsystem identifier. It can be up to four characters long. All characters must be alphanumeric (uppercase A through Z, 0 through 9), it must start with an alphabetic character. The queue manager will have the same name as the subsystem, therefore you can use only characters that are allowed for both z/OS subsystem names and IBM MQ object names.

cpf

The command prefix string (see “Defining command prefix strings (CPFs)” on page 944 for information about CPFs).

scope

The system scope, used if you are running in a z/OS sysplex (see “CPFs in a sysplex environment” on page 945 for information about system scope).

Figure 97 on page 944 shows several examples of IEFSSNss statements.

```
CSQ1,CSQ3INI, 'CSQ3EPX,+mqs1cpf,S'
CSQ2,CSQ3INI, 'CSQ3EPX,+mqs2cpf,S'
CSQ3,CSQ3INI, 'CSQ3EPX,++,S'
```

Figure 97. Sample IEFSSNss statements for defining subsystems

Note: When you have created objects in a subsystem, you cannot change the subsystem name or use the page sets from one subsystem in another subsystem. To do either of these, you must unload all the objects and messages from one subsystem and reload them into another.

Table 55 on page 944 gives a number of examples showing the associations of subsystem names and command prefix strings (CPFs), as defined by the statements in Figure 97 on page 944.

Table 55. Subsystem name to CPF associations

IBM MQ subsystem name	CPF
CSQ1	+mqs1cpf
CSQ2	+mqs2cpf
CSQ3	++

Note: The ACTIVATE and DEACTIVATE functions of the z/OS command SETSSI are not supported by IBM MQ.

To check the status of the changes, issue the following command in SDSF: /D SSI , L. You will see the new subsystems created with ACTIVE status.

z/OS Defining command prefix strings (CPFs)

Each subsystem instance of IBM MQ can have a command prefix string to identify that subsystem.

Adopt a system-wide convention for your CPFs for all subsystems to avoid conflicts. Adhere to the following guidelines:

- Define a CPF as string of up to eight characters.
- Do not use a CPF that is already in use by any other subsystem, and avoid using the JES backspace character defined on your system as the first character of your string.
- Define your CPF using characters from the set of valid characters listed in Table 57 on page 945.

- Do not use a CPF that is an abbreviation for an already defined process or that might be confused with command syntax. For example, a CPF such as 'D' conflicts with z/OS commands such as DISPLAY. To avoid this happening, use one of the special characters (shown in [Table 57 on page 945](#)) as the first or only character in your CPF string.
- Do not define a CPF that is either a subset or a superset of an existing CPF. For an example, see [Table 56 on page 945](#).

Table 56. Example of CPF subset and superset rules

Subsystem name	CPF defined	Commands routed to
MQA	!A	MQA
MQB	!B	MQB
MQC1	!C1	MQC1
MQC2	!C2	MQC2
MQB1	!B1	MQB

Commands intended for subsystem MQB1 (using CPF !B1) are routed to subsystem MQB because the CPF for this subsystem is !B, a subset of !B1. For example, if you entered the command:

```
!B1 START QMGR
```

subsystem MQB receives the command:

```
1 START QMGR
```

(which, in this case, it cannot deal with).

You can see which prefixes exist by issuing the z/OS command DISPLAY OPDATA.

If you are running in a sysplex, z/OS diagnoses any conflicts of this type at the time of CPF registration (see [“CPF in a sysplex environment” on page 945](#) for information about CPF registration).

[Table 57 on page 945](#) shows the characters that you can use when defining your CPF strings:


Table 57. Valid character set for CPF strings

Character set	Contents
Alphabetic	Uppercase A through Z, lowercase a through z
Numeric	0 through 9
National (see note)	@ \$ # (Characters that can be represented as hexadecimal values)
Special	. [] * & + - = € < ! ; % _ ? : >

Note:

The system recognizes the following hexadecimal representations of the national characters: @ as X'7C', \$ as X'5B', and # as X'7B'. In countries other than the U.S., the U.S. national characters represented on terminal keyboards might generate a different hexadecimal representation and cause an error. For example, in some countries the \$ character might generate an X'4A'.

The semicolon (;) is valid as a CPF but on most systems, this character is the command delimiter.

 *CPFs in a sysplex environment*

Use this topic to understand how to use CPFs within the scope of a sysplex.

If used in a sysplex environment, IBM MQ registers your CPFs to enable you to enter a command from any console in the sysplex and route that command to the appropriate system for execution. The command responses are returned to the originating console.

Defining the scope for sysplex operation

Scope is used to determine the type of CPF registration performed by the IBM MQ subsystem when you are running IBM MQ in a sysplex environment.

Possible values for scope are as follows:

M

System scope.

The CPF is registered with z/OS at system IPL time by IBM MQ and remains registered for the entire time that the z/OS system is active.

IBM MQ commands must be entered at a console connected to the z/OS image running the target subsystem, or you must use ROUTE commands to direct the command to that image.

Use this option if you are not running in a sysplex.

S

Sysplex started scope.

The CPF is registered with z/OS when the IBM MQ subsystem is started, and remains active until the IBM MQ subsystem terminates.

You must use ROUTE commands to direct the original START QMGR command to the target system, but all further IBM MQ commands can be entered at any console connected to the sysplex, and are routed to the target system automatically.

After IBM MQ termination, you must use the ROUTE commands to direct subsequent START commands to the target IBM MQ subsystem.

X

Sysplex IPL scope.

The CPF is registered with z/OS at system IPL time by IBM MQ and remains registered for the entire time that the z/OS system is active.

IBM MQ commands can be entered at any console connected to the sysplex, and are routed to the image that is executing the target system automatically.

An IBM MQ subsystem with a CPF with scope of S can be defined on one or more z/OS images within a sysplex, so these images can share a single subsystem name table. However, you must ensure that the initial START command is issued on (or routed to) the z/OS image on which you want the IBM MQ subsystem to run. If you use this option, you can stop the IBM MQ subsystem and restart it on a different z/OS image within the sysplex without having to change the subsystem name table or perform an IPL of a z/OS system.

An IBM MQ subsystem with a CPF with scope of X can only be defined on one z/OS image within a sysplex. If you use this option, you must define a unique subsystem name table for each z/OS image requiring IBM MQ subsystems with CPFs of scope X.

If you want to use the z/OS automatic restart manager (ARM) to restart queue managers in different z/OS images automatically, every queue manager must be defined in each z/OS image on which that queue manager might be restarted. Every queue manager must be defined with a sysplex-wide, unique 4-character subsystem name with a CPF scope of S.

Create procedures for the IBM MQ queue manager

Each IBM MQ subsystem needs a cataloged procedure to start the queue manager. You can create your own or use the IBM-supplied procedure library.

- Repeat this task for each IBM MQ queue manager.
- You might need to modify the cataloged procedure when migrating from a previous version.

For each IBM MQ subsystem defined in the subsystem name table, create a cataloged procedure in a procedure library for starting the queue manager. The IBM-supplied procedure library is called SYS1.PROCLIB, but your installation might use its own naming convention.

The name of the queue manager started task procedure is formed by concatenating the subsystem name with the characters MSTR. For example, subsystem CSQ1 has the procedure name CSQ1MSTR. You need one procedure for each subsystem you define.

You need to include the library containing messages in your selected language:

- thlqual.SCSQSNLE, for US English mixed case
- thlqual.SCSQSNU, for US English uppercase
- thlqual.SCSQSNLK, for Japanese
- thlqual.SCSQSNLF, for French
- thlqual.SCSQSNU, for Chinese

Many examples and instructions in this product documentation assume that you have a subsystem called CSQ1. You might find these examples easier to use if a subsystem called CSQ1 is created initially for installation verification and testing purposes.

Two sample started task procedures are provided in thlqual.SCSQPROC. Member CSQ4MSTR uses one page set for each class of message, member CSQ4MSRR uses multiple page sets for the major classes of message. Copy one of these procedures to member xxxxMSTR (where xxxx is the name of your IBM MQ subsystem) of your SYS1.PROCLIB or, if you are not using SYS1.PROCLIB, your procedure library. Copy the sample procedure to a member in your procedure library for each IBM MQ subsystem that you define.

When you have copied the members, you can tailor them to the requirements of each subsystem, using the instructions in the member. For information about specifying limits of storage used by the queue manager, see [Storage configuration](#). You can also use symbolic parameters in the JCL to allow the procedure to be modified when it is started. If you have several IBM MQ subsystems, you might find it advantageous to use JCL include groups for the common parts of the procedure, to simplify future maintenance.

If you are using queue sharing groups, the STEPLIB concatenation must include the Db2 runtime target library SDSNLOAD, and it must be APF-authorized. This library is only required in the STEPLIB concatenation if it is not accessible through the link list or LPA.

Notes:

1. You can make a note of the names of your bootstrap data set (BSDS), logs, and page sets for use in JCL and then define these sets at a later step in the process.
2. Sample started task procedures CSQ4MSTR and CSQ4MSRR have been updated to include, but leave commented out, the CSQMINI DD card that can be used to define a QMINI data set that contains transport security, that is, SSL or TLS properties.

You can use [“The QMINI data set” on page 953](#) to enable or disable TLS 1.3 support and/or be used to define a custom list of CipherSpecs to be used by channels.

Related concepts

[“Create procedures for the channel initiator” on page 947](#)

For each IBM MQ subsystem, tailor a copy of CSQ4CHIN. Depending on what other products you are using, you might need to allow access to other data sets.

Create procedures for the channel initiator

For each IBM MQ subsystem, tailor a copy of CSQ4CHIN. Depending on what other products you are using, you might need to allow access to other data sets.

- Repeat this task for each IBM MQ queue manager.
- You might need to modify the cataloged procedure when migrating from a previous version.

You need to create a channel-initiator started task procedure for each IBM MQ subsystem that is going to use distributed queuing.

To do this:

1. Copy the sample started task procedure `thlqual.SCSQPROC(CSQ4CHIN)` to your procedure library. Name the procedure `xxxxx CHIN`, where `xxxxx` is the name of your IBM MQ subsystem (for example, `CSQ1CHIN` would be the channel initiator started task procedure for queue manager `CSQ1`).
2. Make a copy for each IBM MQ subsystem that you are going to use.
3. Tailor the procedures to your requirements using the instructions in the sample procedure `CSQ4CHIN`. You can also use symbolic parameters in the JCL to allow the procedure to be modified when it is started. This is described with the start options in [Administering IBM MQ for z/OS](#).

Concatenate the distributed queuing library `thlqual.SCSQMVR1`.

Access to the LE runtime library `SCEERUN` is required; if it is not in your link list (`SYS1.PARMLIB(LNKLSTkk)`), concatenate it in the `STEPLIB DD` statement.

V9.4.0 Consider adjusting the `MEMLIMIT` parameter using the information in [Storage configuration](#).

4. Authorize the procedures to run under your external security manager.
5. You need to include the library containing messages in your selected language:
 - `thlqual.SCSQSNLE`, for US English mixed case
 - `thlqual.SCSQSNLU`, for US English uppercase
 - `thlqual.SCSQSNLK`, for Japanese
 - `thlqual.SCSQSNLF`, for French
 - `thlqual.SCSQSNLC`, for Chinese

The channel initiator is a long running address space. To prevent its termination after a restricted amount of CPU has been consumed, confirm that either:

- The default for started tasks in your z/OS system is unlimited CPU; a JES2 configuration statement for `JOBCLASS(STC)` with `TIME=(1440,00)` achieves this, or
- Explicitly add a `TIME=1440`, or `TIME=NOLIMIT`, parameter to the `EXEC` statement for `CSQXJST`.

You can add the exit library (`CSQXLIB`) to this procedure later if you want to use channel exits. You need to stop and restart your channel initiator to do this.

If you are using TLS, access to the system TLS runtime library is required. This library is called `SIEALNKE`. The library must be APF authorized.

If you are using TCP/IP, the channel initiator address space must be able to access the `TCPIP.DATA` data set that contains TCP/IP system parameters. The ways that the data set has to be set up depends on which TCP/IP product and interface you are using. They include:

- Environment variable, `RESOLVER_CONFIG`
- `/etc/resolv.conf` on the file system
- `//SYSTCPD DD` statement
- `//SYSTCPDD DD` statement
- `jobname/userid.TCPIP.DATA`
- `SYS1.TCPPARMS(TCPDATA)`
- `zapname.TCPIP.DATA`

Some of these affect your started-task procedure JCL. For more information, see [z/OS Communications Server: IP Configuration Guide](#).

Related concepts

[“Define the IBM MQ subsystem to a z/OS WLM service class” on page 949](#)

To give IBM MQ appropriate performance priority in the z/OS system, you must assign the queue manager and channel initiator address spaces to an appropriate z/OS workload management (WLM) service class. If you do not do this explicitly, inappropriate defaults might apply.

Define the IBM MQ subsystem to a z/OS WLM service class

To give IBM MQ appropriate performance priority in the z/OS system, you must assign the queue manager and channel initiator address spaces to an appropriate z/OS workload management (WLM) service class. If you do not do this explicitly, inappropriate defaults might apply.

- *Repeat this task for each IBM MQ queue manager.*
- *You do not need to perform this task when migrating from a previous version.*

Use the ISPF dialog supplied with WLM to perform the following tasks:

- Extract the z/OS WLM policy definition from the WLM couple data set.
- Update this policy definition by adding queue manager and channel initiator started task procedure names to the chosen service class
- Install the changed policy on the WLM couple data set

Then activate this policy using the z/OS command

```
V WLM,POLICY=policyname,REFRESH
```

See [Planning your IBM MQ environment on z/OS](#) for more information on setting performance options.

Related concepts

[“Set up the Db2 environment” on page 986](#)

If you are using queue sharing groups you must create the required Db2 objects by customizing and running a number of sample jobs.

Implement your ESM security controls

Implement security controls for queue managers and the channel initiator.

- *Repeat this task for each IBM MQ queue manager.*
- *You might need to perform this task when migrating from a previous version.*

If you use RACF® as your external security manager, see [Setting up security on z/OS](#), which describes how to implement these security controls.

If you are using the channel initiator, you must also do the following:

- If your subsystem has connection security active, define a connection security profile ssid.CHIN to your external security manager (see [Connection security profiles for the channel initiator](#) for information about this).
- If you are using Transport Layer Security (TLS) or a sockets interface, ensure that the user ID under whose authority the channel initiator is running is configured to use z/OS UNIX System Services, as described in the [z/OS UNIX System Services Planning](#) documentation.
- If you are using TLS, ensure that the user ID under whose authority the channel initiator is running is configured to access the key ring specified in the SSLKEYR parameter of the ALTER QMGR command.

Before you start the queue manager, set up IBM MQ data set and system security by:

- Authorizing the queue manager started task procedure to run under your external security manager.
- Authorizing access to the queue manager data sets.
- Configuring z/OS data set encryption if required.

See the section, [confidentiality for data at rest on IBM MQ for z/OS with data set encryption](#), for more information.

For details about how to do this, see [Security installation tasks for z/OS](#).

If you are using RACF, provided you use the RACF STARTED class, you do not need to perform an IPL of your system (see [RACF authorization of started-task procedures](#)).

Related concepts

[“Update SYS1.PARMLIB members” on page 950](#)

To ensure that your changes remain in effect after an IPL, you must update some members of SYS1.PARMLIB

[“Implement ESM security controls for the queue sharing group” on page 990](#)

Implement security controls for all queue managers in a queue sharing group, to access Db2 and the coupling facility list structures.

Update SYS1.PARMLIB members

To ensure that your changes remain in effect after an IPL, you must update some members of SYS1.PARMLIB

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *If you are using queue sharing groups, you must ensure that the settings for IBM MQ are identical on each z/OS system in the sysplex.*
- *You might need to perform this task when migrating from a previous version.*

Update SYS1.PARMLIB members as follows:

1. Update member IEFSSNss as described in [“Define the IBM MQ subsystem to z/OS” on page 943](#).
2. Change IEASYSpp so that the following members are used when an IPL is performed:
 - the PROGxx or IEAAPFaa members used in [“La fonction APF autorise les bibliothèques de chargement IBM MQ” on page 937](#)
 - the LNKLSTkk and LPALSTmm members used in [“Mise à jour de la liste de liens z/OS et LPA” on page 938](#)
 - the SCHEDxx member used in [“Update the z/OS program properties table” on page 942](#)
 - the IEFSSNss member used in [“Define the IBM MQ subsystem to z/OS” on page 943](#)

Related concepts

[“Customize the initialization input data sets” on page 950](#)

Make working copies of the sample initialization input data sets and tailor them to suit your system requirements.

Customize the initialization input data sets

Make working copies of the sample initialization input data sets and tailor them to suit your system requirements.

- *Repeat this task for each IBM MQ queue manager.*
- *You need to perform this task when migrating from a previous version.*

Each IBM MQ queue manager gets its initial definitions from a series of commands contained in the IBM MQ *initialization input data sets*. These data sets are referenced by the DD names CSQINP1, CSQINP2, and CSQINPT defined in the queue manager started task procedure.

Responses to these commands are written to the initialization output data sets referenced by the DD names CSQOUT1, CSQOUT2 and CSQOUTT.

To preserve the originals, make working copies of each sample. Then you can tailor the commands in these working copies to suit your system requirements.

If you use more than one IBM MQ subsystem, if you include the subsystem name in the high-level qualifier of the initialization input data set name, you can identify the IBM MQ subsystem associated with each data set more easily.

Refer to the following topics for further information about the samples:

- [Initialization data set formats](#)
- [Using the CSQINP1 sample](#)
- [Using the CSQINP2 samples](#)
- [Using the CSQINPX sample](#)
- [Using the CSQINPT sample](#)

Initialization data set formats

The initialization input data sets can be partitioned data set (PDS) members or sequential data sets. They can be a concatenated series of data sets. Define them with a record length of 80 bytes, where:

- Only columns 1 through 72 are significant. Columns 73 through 80 are ignored.
- Records with an asterisk (*) in column 1 are interpreted as comments and are ignored.
- Blank records are ignored.
- Each command must start on a new record.
- A trailing - means continue from column 1 of the next record.
- A trailing + means continue from the first non-blank column of the next record.
- The maximum number of characters permitted in a command is 32 762.

The initialization output data sets are sequential data sets, with a record length of 125, a record format of VBA, and a block size of 629.

Using the CSQINP1 sample

Data set th1qua1 .SCSQPROC holds two members which contain definitions of buffer pools, page set to buffer pool associations, and an ALTER SECURITY command.

Member CSQ4INP1 uses one page set for each class of message. The messages are divided into the following classes:

- System-related messages.
- Important long-lived messages.
- Short-lived messages.
- Miscellaneous messages.

Member CSQ4INPR uses multiple page sets for each of the major classes of message, and one page set for each other class. The following are the major classes of messages:

- Important long-lived messages.
- Short-lived messages.

Include the appropriate sample in the CSQINP1 concatenation of your queue manager started task procedure.

Notes:

1. IBM MQ supports up to 100 buffer pools in the range zero through 99. The DEFINE BUFFPOOL command can only be issued from a CSQINP1 initialization data set. The definitions in the sample specify four buffer pools.
2. Each page set used by the queue manager must be defined in the CSQINP1 initialization data set by using the DEFINE PSID command. The page set definition associates a buffer pool ID with a page set. If no buffer pool is specified, buffer pool zero is used by default.

Page set zero (00) must be defined. It contains all the object definitions. You can define up to 100 page sets for each queue manager.

3. The ALTER SECURITY command can be used to alter the security attributes TIMEOUT and INTERVAL. In CSQ4INP1, the default values are defined as 54 for TIMEOUT and 12 for INTERVAL.

See [Planning your page sets and buffer pools](#) for information about organizing buffer pools and page sets.

If you change the buffer pool and page set definitions dynamically while the queue manager is running, you should also update the CSQINP1 definitions. The changes are only retained for a cold start of IBM MQ, unless the buffer pool definition includes the REPLACE attribute.

Using the CSQINP2 samples

This table lists the members of th1qua1 .SCSQPROC that can be included in the CSQINP2 concatenation of your queue manager started task procedure, with a description of their function. The naming convention is CSQ4IN*. CSQ4INY* members should be modified for your configuration. You should avoid changing CSQINS* members because you will need to reapply any changes when you migrate to the next release. Instead, you can put DEFINE or ALTER commands in CSQ4INY* members.

Member name	Description
CSQ4INSG	System object definitions.
CSQ4INSA	System object and default rules for channel authentication.
CSQ4INSX	System object definitions.
CSQ4INSS	Customize and include this member if you are using queue sharing groups.
CSQ4INSJ	Customize and include this member if you are using publish/subscribe using JMS.
CSQ4INSM	System object definitions for Advanced Message Security.
CSQ4INSR	Customize and include this member if you are using WebSphere Application Server, or the queued publish/subscribe interface supported by the queued publish/subscribe daemon in IBM MQ.
CSQ4DISP	CSQINP2 sample for displaying object definitions.
CSQ4INYC	Clustering definitions.
CSQ4INYD	Distributed queuing definitions.
CSQ4INYG	General definitions.
CSQ4INYR	Storage class definitions, using multiple page sets for the major classes of message.
CSQ4INYS	Storage class definitions, using one page set for each class of message.

You need to define objects once only, not each time that you start a queue manager, so it is not necessary to include these definitions in CSQINP2 every time. If you do include them every time, you are attempting to define objects that already exist, and you will get messages similar to the following:

```
CSQM095I +CSQ1 CSQMAQLC QLOCAL(SYSTEM.DEFAULT.LOCAL.QUEUE) ALREADY EXISTS
CSQM090E +CSQ1 CSQMAQLC FAILURE REASON CODE X'00D44003'
CSQ9023E +CSQ1 CSQMAQLC ' DEFINE QLOCAL' ABNORMAL COMPLETION
```

The objects are not damaged by this failure. If you want to leave the SYSTEM definitions data set in the CSQINP2 concatenation, you can avoid the failure messages by specifying the REPLACE attribute against each object.

Using the CSQINPX sample

Sample `thlqual.SCSQPROC(CSQ4INPX)` contains a set of commands that you might want to execute each time the channel initiator starts. These are typically channel-related commands such as `START LISTENER`, which are required every time the channel initiator starts, rather than whenever the queue manager starts, and which are not allowed in the input data sets `CSQINP1` or `CSQINP2`. You must customize this sample before use; you can then include it in the `CSQINPX` data set for the channel initiator.

The IBM MQ commands contained in the data set are executed at the end of channel initiator initialization, and output is written to the data set specified by the `CSQOUTX` DD statement. The output is like that produced by the `COMMAND` function of the IBM MQ utility program (`CSQUTIL`). See [Using the CSQUTIL utility for IBM MQ for z/OS](#).

You can specify any of the IBM MQ commands that can be issued from `CSQUTIL`, not only the channel commands. You can enter commands from other sources while `CSQINPX` is being processed. All commands are issued in sequence, regardless of the success of the previous command.

To specify a command response time, you can use the pseudo-command `COMMAND` as the first command in the data set. This takes a single optional keyword `RESPTIME(nnn)`, where `nnn` is the time, in seconds, to wait for a response to each command. This is in the range 5 through 999; the default is 30.

If IBM MQ detects that the responses to four commands have taken too long, processing of `CSQINPX` is stopped and no further commands are issued. The channel initiator is not stopped, but message [CSQU052E](#) is written to the `CSQOUTX` data set, and message [CSQU013E](#) is sent to the console.

When IBM MQ has completed processing of `CSQINPX` successfully, message [CSQU012I](#) is sent to the console.

Using the CSQINPT sample

This table lists the members of `thlqual.SCSQPROC` that can be included in the `CSQINPT` concatenation of your queue manager started task procedure, with a description of their function.

Member name	Description
<code>CSQ4INST</code>	System default subscription definition.
<code>CSQ4INYT</code>	Publish/Subscribe definitions.

The IBM MQ commands contained in the data set are executed when publish/subscribe initialization completes, and output is written to the data set specified by the `CSQOUTT` DD statement. The output is like that produced by the `COMMAND` function of the IBM MQ utility program (`CSQUTIL`). See [Using the CSQUTIL utility for IBM MQ for z/OS](#).

Related concepts

[“Create the bootstrap and log data sets” on page 955](#)

Use the supplied program `CSQJU003` to prepare the bootstrap data sets (BSDSs) and log data sets.

The QMINI data set

You can use the `QMINI` data set to specify properties that are to be read and processed during queue manager initialization.

Characteristics of the QMINI data set

The `QMINI` data set is a sequential data set, or a member of a partitioned data set, with a maximum record length of 80 bytes (72 bytes for data and eight bytes for the line number).

The following example shows the properties for a sequential `QMINI` data set. Some properties are, of course, based on your environment.

```

Data Set Name . . . . : QM01.QMINI
General Data
Management class . . : STANDARD      Current Allocation
Storage class . . . . : STANDARD      Allocated tracks . : 1
Volume serial . . . . : P5P21E       Allocated extents . : 1
Device type . . . . . : 3390
Data class . . . . . : **None**
Organization . . . . : PS             Current Utilization
Record format . . . . : FB            Used tracks . . . . : 0
Record length . . . . : 80           Used extents . . . . : 0
Block size . . . . . : 3120
1st extent tracks . . : 1
Secondary tracks . . . : 1
Data set name type . . :
Data set encryption . : NO
SMS Compressible . . . : NO
Dates
Creation date . . . . : 2020/08/11
Referenced date . . . : ***None***
Expiration date . . . : ***None***

```

thlqual.SCSQPROC, includes:

- The sample contents for a QMINI data set in CSQ4QMIN.
- An example of specifying the QMINI data set using the //CSQMINI DD card, in the queue manager startup JCL, in the started task procedures CSQ4MSTR and CSQ4MSRR.

Notes:

- The code that parses the data set only parses the first 72 bytes of each record.
- Line numbers are ignored so it is not necessary to specify line numbers.
- If a line starts with an asterisk character (*), the line is treated as a comment.
- The contents of the QMINI data set are parsed during queue manager startup. If the contents are parsed successfully, message CSQM578I is issued in the queue manager job log. If any errors are encountered during parsing, error messages, for example CSQM573E, are issued in the queue manager job log but the queue manager still starts.

Check for error messages, and resolve any issues in the contents of the QMINI data set.

If the queue manager is unable to parse the QMINI data set, you can start the channel initiator, but you cannot start any channels that are configured to use SSL or TLS as the security configuration settings are unknown.

- If you make any updates to the data set after you have started the queue manager, you must restart the queue manager to pick up the changes.

The TransportSecurity stanza

From IBM MQ for z/OS 9.2.0, the QMINI data set supports the TransportSecurity stanza. This stanza provides similar function to that provided by the SSL stanza in the qm.ini file on IBM MQ for Multiplatforms.

The TransportSecurity stanza supports the following properties:

AllowTLSV13

Whether a queue manager is able to use the TLS 1.3 CipherSpecs; valid values are: *TRUE/T/YES/Y* or *FALSE/F/NO/N*.

For migrated queue managers, TLS 1.3 is not enabled by default. You can enable TLS 1.3 by defining a QMINI data set with the TransportSecurity stanza and **AllowTLSV13=TRUE**.

For newly created queue managers TLS 1.3 is enabled by default.

AllowedCipherSpecs

A custom list of CipherSpecs that are enabled.

See [Providing a custom list of ordered and enabled CipherSpecs on IBM MQ for z/OS](#) for more information on this property.

Duplicate CipherSpec names in the list are ignored.

OutboundSNI

Whether the Server Name Indication (SNI) is set to the target IBM MQ channel name to the remote system when initiating a TLS connection, or to the hostname; valid values are: CHANNEL or HOSTNAME.

If the destination channel is configured with a certificate label on the channel object CERTLABL field, you must set CERTLABL to the channel value. If a connection with a setting of HOSTNAME is made to a channel with a CERTLABL setting, the connection fails and an AMQ9673 message is printed in the remote queue manager error logs.

The following example shows how the TransportSecurity stanza is specified:

```
TransportSecurity:
AllowedCipherSpecs=ECDHE_RSA_AES_128_GCM_SHA256,
                  ECDHE_RSA_AES_256_GCM_SHA384
AllowTLSV13=TRUE
```

Create the bootstrap and log data sets

Use the supplied program CSQJU003 to prepare the bootstrap data sets (BSDSs) and log data sets.

Note:

- Repeat this task for each IBM MQ queue manager.
- If you are using z/OS data set encryption to protect the BSDS or active log data sets, you must configure this option before the data sets are allocated in this step.
- You do not need to perform this task when migrating from a previous version.
- If you are migrating a queue manager and adding z/OS data set encryption for active log data sets or BSDS, you need to convert the data sets.
- For more information about configuring z/OS data set encryption, and converting existing IBM MQ data sets to be encrypted, see [confidentiality for data at rest on IBM MQ for z/OS with data set encryption..](#)

The sample JCL and Access Method Services (AMS) control statements to run CSQJU003 to create a single or dual logging environment are held in thlqual.SCSQPROC(CSQ4BSDS). Customize and run this job to create your BSDSs and logs and to preformat the logs.

Important: You should use the newest version of CSQ4BSDS, or update your JCL manually to use RECORDS(850 60).

The started task procedure, CSQ4MSTR, described in “[Create procedures for the IBM MQ queue manager](#)” on page 946, refers to BSDSs in statements of the form:

```
//BSDS1 DD DSN=++HLQ++.BSDS01,DISP=SHR
//BSDS2 DD DSN=++HLQ++.BSDS02,DISP=SHR
```

The log data sets are referred to by the BSDSs.

Note:

1. The BLKSIZE must be specified on the SYSPRINT DD statement in the LOGDEF step. The BLKSIZE must be 629.
2. To help identify bootstrap data sets and log data sets from different queue managers, include the subsystem name in the high level qualifier of these data sets.
3. If you are using queue sharing groups, you must define the bootstrap and log data sets with SHAREOPTIONS(2 3).

See [Planification sur z/OS](#) for information about planning bootstrap and log data sets and their sizes.

From IBM MQ 8.0, the 8 byte log RBA enhancement improves the availability of a queue manager, as described in [Larger log Relative Byte Address](#). To enable 8 byte log RBA on a queue manager before the queue manager is first started, perform the following steps after creating your logging environment.

Note: For queue managers created at IBM MQ 9.3.0 or later 8 byte log RBA is already enabled, so the following steps are not necessary.

1. Using **IDCAMS ALTER**, rename the version 1 format BSDSs (created using the CSQJU003 program) to something like ++HLQ++ . V1 . BSDS01.

Note: Ensure that you rename the data and index components as well as the VSAM cluster.

2. Allocate new BSDSs with the same attributes as the ones already defined. These will become the version 2 format BSDSs that will be used by the queue manager when it is started.
3. Run the BSDS conversion utility (CSQJUCNV) to convert the version 1 format BSDSs to the new version 2 format BSDSs.
4. Once the conversion completes successfully, delete the version 1 format BSDSs.

Note: If the queue manager is in a queue sharing group, all queue managers in the queue sharing group must have been started as follows before 8 byte log RBA can be enabled:

- If the queue manager is at IBM MQ 9.0.0 LTS it must have been started with **OPMODE(NEWFUNC,900)** or **OPMODE(NEWFUNC,800)**
- If the queue manager is at IBM MQ 9.0.n CD, or IBM MQ 9.1.0 LTS, or later, it needs to have been started at that level

Related concepts

[“Define your page sets” on page 956](#)

Define page sets for each queue manager using one of the supplied samples.

Define your page sets

Define page sets for each queue manager using one of the supplied samples.

- Repeat this task for each IBM MQ queue manager.

If you using z/OS data set encryption to protect the page sets, you must configure this option before the data sets are allocated in this step.

- You do not need to perform this task when migrating from a previous version.

If you are migrating a queue manager and adding z/OS data set encryption for page sets, you need to convert the page sets.

See the section, [Confidentiality for data at rest on IBM MQ for z/OS with data set encryption](#). for more information about configuring z/OS data set encryption and converting existing IBM MQ data sets to be encrypted.

Define separate page sets for each IBM MQ queue manager. thlqual.SCSQPROC(CSQ4PAGE) and thlqual.SCSQPROC(CSQ4PAGR) contain JCL and z/OS access method services (AMS) control statements to define and format page sets. Member CSQ4PAGE uses one page set for each class of message, member CSQ4PAGR uses multiple page sets for the major classes of message. The JCL runs the supplied utility program CSQUTIL. Review the samples and customize them for the number of page sets you want and the sizes to use. See [Planning your page sets and buffer pools](#) for information about page sets and how to calculate suitable sizes.

The started task procedure CSQ4MSTR described in [“Create procedures for the IBM MQ queue manager” on page 946](#) refers to the page sets, in a statement of the form:

```
//CSQP00nn DD DISP=OLD,DSN=xxxxxxxx
```

where *nn* is the page set number between 00 and 99, and *xxxxxxxx* is the data set that you define.

Note:

1. If you intend to use the dynamic page set expansion feature, ensure that secondary extents are defined for each page set. thlqual.SCSQPROC(CSQ4PAGE) shows how to do this.

- To help identify page sets from different queue managers, include the subsystem name in the high level qualifier of the data set associated with each page set.
- If you intend to allow the FORCE option to be used with the FORMAT function of the utility program CSQUTIL, you must add the REUSE attribute on the AMS DEFINE CLUSTER statement.

See the [Optional Parameters](#) section of the z/OS DEFINE CLUSTER command for more information on REUSE.

- If your page sets are to be larger than 4 GB you must use the Storage Management System (SMS) EXTENDED ADDRESSABILITY function.

Related concepts

[“Add the IBM MQ entries to the Db2 tables” on page 989](#)

If you are using queue sharing groups, run the CSQ5PQSG utility to add queue sharing group and queue manager entries to the IBM MQ tables in the Db2 data-sharing group.

Tailor your system parameter module

The IBM MQ system parameter module controls the logging, archiving, tracing, and connection environments that IBM MQ uses in its operation. A default module is supplied. You should create your own system parameter module as some parameters, for example data set names, are usually site specific.

- Repeat this task for each IBM MQ queue manager, as required.
- You might need to perform this task when migrating from a previous version. For details, see [Migrating IBM MQ on z/OS](#).
- To enable Advanced Message Security for z/OS on an existing queue manager, you only need to set SPLCAP to YES as described in [“Utilisation de CSQ6SYSP” on page 959](#). If you are configuring this queue manager for the first time, complete the whole of this task.

The system parameter module has four macros as follows:

Macro name	Purpose
CSQ6SYSP	Specifies the connection and tracing parameters, see “Utilisation de CSQ6SYSP” on page 959
CSQ6LOGP	Controls log initialization, see “Using CSQ6LOGP” on page 969
CSQ6ARVP	Controls archive initialization, see “Using CSQ6ARVP” on page 973
CSQ6USGP	Controls usage recording, see “Using CSQ6USGP” on page 980

IBM MQ supplies a default system parameter module, CSQZPARM, which is invoked automatically if you issue the START QMGR command (without a PARM parameter) to start an instance of IBM MQ. CSQZPARM is in the APF-authorized library thlqual.SCSQAUTH also supplied with IBM MQ. The values of these parameters are displayed as a series of messages when you start IBM MQ.

See [START QMGR](#) for more information about how this command is used.

Creating your own system parameter module

If CSQZPARM does not contain the system parameters you want, you can create your own system parameter module using the sample JCL provided in thlqual.SCSQPROC(CSQ4ZPRM).

To create your own system parameter module:

- Make a working copy of the JCL sample.
- Edit the parameters for each macro in the copy as required. If you remove any parameters from the macro calls, the default values are automatically picked up at run time.

3. Replace the placeholder ++NAME++ with the name that the load module is to take (this can be CSQZPARM).
4. If your assembler is not high-level assembler, change the JCL as required by your assembler.
5. Run the JCL to assemble and link edit the tailored versions of the system parameter macros to produce a load module. This is the new system parameter module with the name that you have specified.
6. Put the load module produced in an APF-authorized user library.
7. Add user READ access to the APF-authorized user library.
8. Include this library in the IBM MQ queue manager started task procedure STEPLIB. This library name must come before the library thlqual.SCSQAUTH in STEPLIB.
9. Invoke the new system parameter module when you start the queue manager. For example, if the new module is named NEWMODS, issue the command:

```
START QMGR PARM(NEWMODS)
```

10. Ensure successful completion of the command by checking the job log. There should be an entry in the log similar to the following:

```
CSQ9022I CDL1 CSQYASCP 'START QMGR' NORMAL COMPLETION
```

You can also specify the parameter module name in the queue manager startup JCL. For more information, see [Using MQSC to start and stop a queue manager on z/OS](#).

Note: If you choose to name your module CSQZPARM, you do not need to specify the PARM parameter on the START QMGR command.

Fine tuning a system parameter module

IBM MQ also supplies a set of three assembler source modules, which can be used to fine-tune an existing system parameter module. These modules are in library thlqual.SCSQASMS. Typically, you use these modules in a test environment to change the default parameters in the system parameter macros. Each source module calls a different system parameter macro:

This assembler source module...	Calls this macro...
CSQFSYSP	CSQ6SYSP (connection and tracing parameters)
CSQJLOGP	CSQ6LOGP (log initialization)
CSQJARVP	CSQ6ARVP (archive initialization)

This is how you use these modules:

1. Make working copies of each assembler source module in a user assembler library.
2. Edit your copies by adding or altering the values of any parameters as required.
3. Assemble your copies of any edited modules to create object modules in a user object library.
4. Link edit these object code modules with an existing system parameter module to produce a load module that is the new system parameter module.
5. Ensure that new system parameter module is a member of a user authorized library.
6. Include this library in the queue manager started task procedure STEPLIB. This library must come before the library thlqual.SCSQAUTH in STEPLIB.
7. Invoke the new system parameter module by issuing a START QMGR command, specifying the new module name in the PARM parameter, as before.

A sample usermod is provided in member CSQ4UZPR of SCSQPROC which demonstrates how to manage customized system parameters under SMP/E control.

Altering system parameters

You can alter some system parameters while a queue manager is running; see the [SET SYSTEM](#), [SET LOG](#), and [SET ARCHIVE](#) commands.

Put the SET commands in your initialization input data sets so that they take effect every time you start the queue manager.

Related concepts

“Tailor the channel initiator parameters” on page 981

Use ALTER QMGR to customize the channel initiator to suit your requirements.

Utilisation de CSQ6SYSP

Utilisez cette rubrique comme référence pour définir les paramètres système à l'aide de CSQ6SYSP.

Les paramètres par défaut de CSQ6SYSP et la possibilité de modifier chaque paramètre à l'aide de la commande SET SYSTEM sont présentés dans [Tableau 60](#), à la [page 959](#). Si vous souhaitez modifier l'une de ces valeurs, consultez les descriptions détaillées des paramètres.

Paramètre	Description	Valeur par défaut	commande set
«ACCTIME» , à la page 961	Durée, en minutes et en secondes, entre chaque collecte de données comptables.	-1	✓
«ACELIM» , à la page 961	Taille du pool de stockage ACE en blocs de 1 Ko.	0 (pas de limite)	✓
«CACHE» , à la page 962	Indique le type de cache de cluster à utiliser.	STATIQUE	-
«CMDUSER» , à la page 962	ID utilisateur par défaut pour les contrôles de sécurité de commande.	CSQOPR	-
«EXCLMSG» , à la page 963	Indique une liste de messages à exclure de tout journal. Les messages de cette liste ne sont pas envoyés à la console z/OS ni au journal papier. Par conséquent, l'utilisation du paramètre EXCLMSG pour exclure des messages est plus efficace du point de vue de l'unité centrale que l'utilisation des méthodes décrites dans «Suppress information messages» , à la page 985	()	✓
«EXITLIM» , à la page 963	Durée maximale (en secondes) pendant laquelle les exits de gestionnaire de files d'attente peuvent s'exécuter lors de chaque appel.	30	-
«EXITTCB» , à la page 963	Nombre de tâches de serveur démarrées à utiliser pour exécuter les exits de gestionnaire de files d'attente.	8	-
«LOGLOAD» , à la page 963	Nombre d'enregistrements de journal écrits par IBM MQ entre le début d'un point de contrôle et le suivant.	500 000	✓

Tableau 60. Valeurs par défaut des paramètres CSQ6SYSP (suite)

Paramètre	Description	Valeur par défaut	commande set
«MULCCAPT» , à la page 964	Détermine la propriété Prix d'utilisation mesuré qui contrôle l'algorithme de collecte des données utilisées par MULC (Measured Usage License Charge).	Voir description des paramètres	-
«Mot de passe à utilisation unique» , à la page 964	Paramètres de connexion OTMA.	Voir description des paramètres	-
«QINDXBLD» , à la page 965	Détermine si le redémarrage du gestionnaire de files d'attente attend que tous les index soient régénérés ou se termine avant que tous les index soient régénérés.	ATTENDRE	-
«QMCCSID» , à la page 965	Identificateur de jeu de caractères codés pour le gestionnaire de files d'attente.	Zéro	-
«DONNEES SGQ» , à la page 965	Paramètres du groupe de partage de files d'attente.	Voir description des paramètres	-
«RESAUDIT» , à la page 966	Paramètre d'audit RESLEVEL.	YES	-
«ROUTCDE» , à la page 966	Code de routage de message affecté aux messages non sollicités à partir d'une console spécifique.	1	-
«SERVICE» , à la page 967	Réservé à l'utilisation par IBM.	0	✓
«SMFACCT» , à la page 967	Indique si les données comptables SMF doivent être collectées lors du démarrage du gestionnaire de files d'attente. Notez que les données de comptabilité de canal de classe 4 sont collectées uniquement lorsque l'initiateur de canal est démarré.	NO	-
SMFSTAT	Indique si les statistiques SMF doivent être collectées lorsque le gestionnaire de files d'attente est démarré. Notez que les données statistiques de l'initiateur de canal de classe 4 sont collectées uniquement lorsque l'initiateur de canal est démarré.	NO	-
SPLCAP	Indique si la fonction de règle de sécurité de file d'attente est activée sur ce gestionnaire de files d'attente. Pour Advanced Message Security for z/OS, définissez ce paramètre sur YES.	NO	-
STATIME	Durée, en minutes et en secondes, entre chaque collecte de statistiques.	30	✓

Tableau 60. Valeurs par défaut des paramètres CSQ6SYSP (suite)

Paramètre	Description	Valeur par défaut	commande set
<u>TRACSTR</u>	Indique si la fonction de trace doit être démarrée automatiquement.	NO	-
<u>TRACTBL</u>	Taille de la table de trace, en blocs de 4 Ko, à utiliser par la fonction de trace globale.	99 (396 Ko)	✓
<u>WLMTIME</u>	Temps écoulé entre l'analyse de l'index de file d'attente pour les files d'attente gérées par WLM.	30	-
<u>WLMTIMU</u>	Unités (minutes ou secondes) pour WLMTIME.	minutes	-

ACCTIME

Indique l'intervalle, en minutes et en secondes, entre les collectes consécutives de données comptables.

Specify a number, either -1, or in the range 0 through 1440 minutes in the format ' mmmm ', or in the range 0 through 1440 minutes, and 0 - 59 seconds, in the format ' mmmm . ss '.

Remarques :

- Lorsque vous spécifiez un intervalle de secondes uniquement, vous devez le préfixer avec la valeur 0. Le plus petit intervalle possible est d'une seconde: '0 . 01'.
- Si vous spécifiez la valeur 0, les données de comptabilité sont collectées à l'intervalle d'enregistrement global SMF. Pour plus d'informations, voir [Utilisation de System Management Facility](#).
- Si vous spécifiez la valeur -1, qui est la valeur par défaut, les données comptables sont collectées à l'intervalle spécifié par la valeur STATIME.

Exemple :

' 0 . 30 ' définit un intervalle de 30 secondes.

' 5 . 30 ' définit un intervalle de 5 minutes et 30 secondes.

' 30 ' définit un intervalle de 30 minutes.

ACELIM

Indique la taille maximale du pool de stockage ACE, par blocs de 1 ko. La valeur doit être comprise entre 0 et 999999. La valeur par défaut zéro indique qu'aucune contrainte n'est imposée, dans les limites de l'espace disponible sur le système.

Vous ne devez attribuer une valeur à ACELIM qu'aux gestionnaires de files d'attente identifiés comme utilisant une quantité exorbitante de mémoire ECSA. Le fait de limiter le pool de stockage d'entrée de contrôle d'accès a pour effet de limiter le nombre de connexions sur le système et par conséquent la quantité de mémoire ECSA utilisée par un gestionnaire de files d'attente.

Une fois que ce dernier a atteint la limite, il n'est plus possible aux applications d'obtenir de nouvelles connexions. L'absence de nouvelles connexions entraîne l'échec du traitement de MQCONN et les applications coordonnées par le biais des services RRS sont susceptibles de subir des défaillances dans une API IBM MQ.

Un ACE représente environ 12,5 % de la ECSA totale requise pour les blocs de contrôle liés aux unités d'exécution pour une connexion. Ainsi, par exemple, la spécification de ACELIM=5120 devrait limiter la quantité totale d'ECSA allouée par le gestionnaire de files d'attente (pour les blocs de contrôle liés aux unités d'exécution) à environ 40960K; , soit 5120 multiplié par 8.

Afin de limiter la quantité totale de mémoire ECSA allouée par le gestionnaire de files d'attente, pour les blocs de contrôle liés aux unités d'exécution à 5120K, une valeur ACELIM de 640 est requise.

Vous pouvez utiliser les enregistrements SMF 115 sous-type 5, générés par la trace CLASS(3) des statistiques pour surveiller la taille du pool de stockage 'ACE/PEB' et définir ainsi une valeur appropriée pour ACELIM.

Vous pouvez obtenir la quantité totale de mémoire ECSA utilisée par le gestionnaire de files d'attente, pour les blocs de contrôle, à partir des enregistrements SMF 115 sous-type 7, écrits par la trace CLASS(2) des statistiques. La quantité totale de stockage ECSA utilisée est la somme des valeurs des zones QSRSPHBGF et QSRSPHBGV.

Pour plus d'informations sur les enregistrements de statistiques SMF 115, voir [Interpreting IBM MQ performance statistics](#).

Envisagez de définir ACELIM comme mécanisme de protection d'une image z/OS contre le mauvais fonctionnement d'un gestionnaire de files d'attente plutôt que comme moyen de contrôler les connexions d'application à un gestionnaire de files d'attente.

CACHE

Indique le type de cache de cluster à utiliser.

Le cache de cluster est une zone de stockage utilisée pour stocker les informations relatives au cluster.

Si le cache du cluster est statique, il a une taille fixe, qui est allouée au démarrage du gestionnaire de files d'attente. Si le cache se remplit, le message CSQM060E est émis et la demande d'application qui nécessitait plus d'espace reçoit une erreur MQRC_CLUSTER_RESOURCE_ERROR.

Si vous affectez à CLCACHE la valeur dynamique, le cache du cluster peut être étendu selon les besoins. Toutefois, vous devez d'abord vous assurer que les exits de charge de travail de cluster installés peuvent fonctionner avec un cache dynamique.

Si un exit de charge de travail de cluster installé ne peut pas fonctionner avec un message de cache dynamique CSQM061E est émis.

MQXCLWLNest fourni pour que les exits de charge de travail de cluster naviguent dans le cache de cluster d'une manière qui fonctionne, que les caches dynamiques ou statiques soient utilisés.

Pour les nouveaux gestionnaires de files d'attente, définissez CLCACHE=DYNAMIC, sauf si vous allez utiliser un exit de charge de travail de cluster qui ne prend pas en charge un cache dynamique.

Pour les gestionnaires de files d'attente existants qui utilisent déjà un cache statique et qui se trouvent dans un cluster auquel peu de nouvelles files d'attente et de nouveaux gestionnaires de files d'attente sont ajoutés, il est raisonnable de continuer à utiliser CLCACHE=STATIC.

Pour les gestionnaires de files d'attente existants qui utilisent déjà un cache statique et qui se trouvent dans un cluster auquel de nombreuses nouvelles files d'attente ou de nouveaux gestionnaires de files d'attente seront ajoutés, commencez à utiliser CLCACHE=DYNAMIC.

STATIQUE

Lorsque le cache de cluster est statique, sa taille est fixée au démarrage du gestionnaire de files d'attente, ce qui est suffisant pour la quantité actuelle d'informations de cluster plus un espace pour l'extension. La taille ne peut pas augmenter tant que le gestionnaire de files d'attente est actif. Il s'agit de l'option par défaut.

DYNAMIQUE

Lorsque le cache du cluster est dynamique, la taille initiale allouée au démarrage du gestionnaire de files d'attente peut être augmentée automatiquement si nécessaire pendant que le gestionnaire de files d'attente est actif.

CMDUSER

Indique l'ID utilisateur par défaut utilisé pour les contrôles de sécurité des commandes. Cet ID utilisateur doit être défini auprès du gestionnaire ESM (par exemple, RACF). Indiquez un nom de 1 à 8 caractères alphanumériques. Le premier caractère doit être alphabétique.

La valeur par défaut est CSQOPR.

EXCLMSG

Indique une liste de messages d'erreur à exclure.

Cette liste est dynamique et mise à jour à l'aide de la commande SET SYSTEM.

La valeur par défaut est une liste vide ().

Les messages sont fournis sans le préfixe CSQ et sans le suffixe du code d'action (I-D-E-A). Par exemple, pour exclure le message CSQX500I, ajoutez X500 à cette liste. Cette liste peut contenir jusqu'à 16 identificateurs de message.

Pour pouvoir être inclus dans la liste, le message doit être émis après le démarrage normal des espaces adresse MSTR ou CHIN et commencer par l'un des caractères suivants: E, H, I, J, L, M, N, P, R, T, V, W, X, Y, 2, 3, 5, 9.

Les identificateurs de message émis à la suite de commandes de traitement peuvent être ajoutés à la liste, mais ils ne seront pas exclus. Par exemple, un identificateur de message est émis suite à la commande DISPLAY USAGE PSID (*), mais ce message ne peut pas être supprimé.

EXITLIM

Indique la durée, en secondes, autorisée pour chaque appel des exits du gestionnaire de files d'attente. (Ce paramètre n'a aucun effet sur les exits de canal.)

Indiquez une valeur comprise entre 5 et 9999.

La valeur par défaut est 30. Le gestionnaire de files d'attente interroge les exits qui s'exécutent toutes les 30 secondes. Lors de chaque interrogation, toutes les demandes qui ont été exécutées pendant une durée supérieure à la durée spécifiée par EXITLIM sont arrêtées de force.

EXITTCB

Indique le nombre de tâches de serveur démarrées à utiliser pour exécuter des exits dans le gestionnaire de files d'attente. (Ce paramètre n'a aucun effet sur les exits de canal.) Vous devez indiquer un nombre au moins aussi élevé que le nombre maximal d'exits (autres que les exits de canal) que le gestionnaire de files d'attente peut avoir à exécuter, sinon il échouera avec une fin anormale 6c6 .

Indiquez une valeur comprise entre zéro et 99. La valeur zéro signifie qu'aucun exit ne peut être exécuté.

La valeur par défaut est 8.

LOGLOAD

Indique le nombre d'enregistrements de journal écrits par IBM MQ entre le début d'un point de contrôle et le suivant. IBM MQ lance un nouveau point de contrôle une fois que le nombre d'enregistrements spécifié a été créé.

Indiquez une valeur comprise entre 200 et 16 000 000.

La valeur par défaut est 500 000.

Plus la valeur est élevée, meilleures sont les performances de IBM MQ ; cependant, le redémarrage prend plus de temps si le paramètre est défini sur une valeur élevée.

Paramètres suggérés:

Système de test	10 000
Système de production	500 000

Dans un système de production, la valeur par défaut fournie peut entraîner une fréquence de point de contrôle trop élevée.

La valeur de LOGLOAD détermine la fréquence des points de contrôle du gestionnaire de files d'attente. Une valeur trop élevée signifie qu'une grande quantité de données est écrite dans le journal entre les points de contrôle, ce qui entraîne une augmentation du temps de redémarrage de la reprise aval du gestionnaire de files d'attente suite à un échec. Une valeur trop faible entraîne des points

de contrôle trop fréquents pendant les pics de charge, ce qui a un impact négatif sur les temps de réponse et l'utilisation du processeur.

Une valeur initiale de 500 000 est suggérée pour LOGLOAD. Pour un débit de messages persistants de 1 ko de 100 messages par seconde (c'est-à-dire 100 MQPUT s avec validation et 100 MQGET s avec validation), l'intervalle entre les points de contrôle est d'environ 5 minutes.

Remarque : Il s'agit d'une recommandation uniquement et la valeur optimale de ce paramètre dépend des caractéristiques du système individuel.

MULCCAPT

Indique l'algorithme à utiliser pour la collecte des données utilisées par la commande MULC (Mesuré Usage License Recharge).

STANDARD

MULC est basé sur l'heure de l'appel MQCONN de l'API IBM MQ à l'heure de l'appel MQDISC de l'API IBM MQ .

Détaillée

MULC est basé sur l'heure entre le début d'un appel d'API IBM MQ et la fin de l'appel d'API IBM MQ .

La valeur par défaut est STANDARD

Mot de passe à utilisation unique

Paramètres OTMA. Ce mot clé prend cinq paramètres positionnels:

OTMACON = (Group, Member, Druexit, Age, Tpipepfx)

Groupe

Il s'agit du nom du groupe XCF auquel appartient cette instance particulière de IBM MQ .

Il peut comporter entre 1 et 8 caractères et doit être entré en majuscules.

La valeur par défaut est vide, ce qui indique que IBM MQ ne doit pas tenter de joindre un groupe XCF.

Membre

Il s'agit du nom de membre de cette instance particulière de IBM MQ au sein du groupe XCF.

Elle peut comporter entre 1 et 16 caractères et doit être entrée en majuscules.

La valeur par défaut est le nom du gestionnaire de files d'attente à 4 caractères.

Sortie de drogue

Indique le nom de l'exit utilisateur de résolution de destination OTMA à exécuter par IMS.

Elle peut comporter entre 1 et 8 caractères.

La valeur par défaut est DFSYDRU0.

Ce paramètre est facultatif ; il est obligatoire si IBM MQ doit recevoir des messages d'une application IMS qui n'a pas été démarrée par IBM MQ. Le nom doit correspondre à l'exit utilisateur de résolution de destination codé dans le système IMS . Pour plus d'informations, voir [«Using OTMA exits in IMS», à la page 1049.](#)

Ancienneté

Représente la durée, en secondes, pendant laquelle un ID utilisateur provenant de IBM MQ est considéré comme précédemment vérifié par IMS.

Elle peut être comprise entre 0 et 2 147 483 647.

La valeur par défaut est 2 147 483 647.

Il est recommandé de définir ce paramètre conjointement avec le paramètre interval de la commande ALTER SECURITY afin de maintenir la cohérence des paramètres de cache de sécurité sur le grand système.

Tpipepfx

Représente le préfixe à utiliser pour les noms Tpipe.

Il comprend trois caractères ; le premier caractère est compris entre A et Z, les caractères suivants sont compris entre A et Z ou entre 0 et 9. La valeur par défaut est CSQ.

Elle est utilisée chaque fois que IBM MQ crée une ressource Tpipe ; le reste du nom est affecté par IBM MQ. Vous ne pouvez pas définir le nom Tpipe complet pour une ressource Tpipe créée par IBM MQ.

QINDEXBLD

Détermine si le redémarrage du gestionnaire de files d'attente attend que tous les index de file d'attente soient régénérés ou se termine avant que tous les index soient régénérés.

ATTENDRE

Le redémarrage du gestionnaire de files d'attente attend que toutes les générations d'index de file d'attente soient terminées. Cela signifie qu'aucune application n'est retardée lors du traitement normal de l'API IBM MQ lors de la création de l'index, car tous les index sont créés avant qu'une application ne puisse se connecter au gestionnaire de files d'attente.

Il s'agit de l'option par défaut.

NOWAIT

Le gestionnaire de files d'attente peut être redémarré avant la fin de la génération de tous les index de file d'attente.

QMCCSID

Indique l'identificateur de jeu de caractères codés par défaut que le gestionnaire de files d'attente (et donc la mise en file d'attente répartie) doit utiliser.

Indiquez une valeur comprise entre zéro et 65535. La valeur doit représenter une page de codes EBCDIC répertoriée comme page de codes z/OS native pour la langue de votre choix dans les [langues nationales](#).

Zéro, qui est la valeur par défaut, signifie utiliser le CCSID actuellement défini ou, si aucun n'est défini, utiliser le CCSID 500. Cela signifie que si vous avez explicitement défini le CCSID sur une valeur différente de zéro, vous ne pouvez pas le réinitialiser en définissant QMCCSID sur zéro ; vous devez maintenant utiliser le CCSID différent de zéro correct. Si QMCCSID est égal à zéro, vous pouvez vérifier quel CCSID est en cours d'utilisation en exécutant la commande DISPLAY QMGR CCSID.

Remarque : Tous les gestionnaires de files d'attente d'un groupe de partage de files d'attente doivent utiliser le même QMCCSID.

DONNEES SGQ

Données du groupe de partage de files d'attente. Ce mot clé prend cinq paramètres positionnels:

QSGDATA = (Qsgname , Dsgname , Db2name , Db2serv , Db2b1ob)

QSGNAME

Il s'agit du nom du groupe de partage de files d'attente auquel appartient le gestionnaire de files d'attente.

Pour connaître les caractères valides, voir [Règles de dénomination des objets IBM MQ](#) . Le nom :

- Peut comporter de 1 à 4 caractères
- Ne doit pas commencer par une valeur numérique
- Ne doit pas se terminer par @.

En effet, pour des raisons d'implémentation, les noms de moins de quatre caractères sont complétés en interne par des symboles @,

La valeur par défaut est vide, ce qui indique que le gestionnaire de files d'attente n'est membre d'aucun groupe de partage de files d'attente.

Nom de fichier

Il s'agit du nom du groupe de partage de données Db2 auquel le gestionnaire de files d'attente doit se connecter.

Il peut comporter entre 1 et 8 caractères et doit être entré en majuscules.

La valeur par défaut est vide, ce qui indique que vous n'utilisez pas de groupes de partage de files d'attente.

Db2name

Il s'agit du nom du sous-système ou de la connexion de groupe Db2 auquel le gestionnaire de files d'attente doit se connecter.

Elle peut comporter de 1 à 4 caractères et doit être entrée en majuscules.

La valeur par défaut est vide, ce qui indique que vous n'utilisez pas de groupes de partage de files d'attente.

Remarque : Le sous-système Db2 (ou la connexion de groupe) doit se trouver dans le groupe de partage de données Db2 spécifié dans le Dsgname, et tous les gestionnaires de files d'attente doivent spécifier le même groupe de partage de données Db2 .

Db2serv

Il s'agit du nombre de tâches serveur utilisées pour accéder à Db2.

Elle peut être comprise entre 4 et 10.

La valeur par défaut est 4.

Db2blob

Il s'agit du nombre de tâches Db2 utilisées pour accéder aux objets BLOB (Binary Large Object).

Elle peut être comprise entre 4 et 10.

La valeur par défaut est 4.

Si vous spécifiez l'un des paramètres de nom (c'est-à-dire **Qsgname**, **Dsgname** ou **Db2name**), vous devez entrer des valeurs pour les autres noms, sinon IBM MQ échoue.

RESAUDIT

Indique si les enregistrements d'audit RACF sont écrits pour les contrôles de sécurité RESLEVEL effectués lors du traitement de la connexion.

Indiquez l'une des valeurs suivantes :

NO

L'audit RESLEVEL n'est pas effectué.

YES

L'audit RESLEVEL est effectué.

La valeur par défaut est YES.

ROUTCDE

Indique le code de routage des messages z/OS par défaut affecté aux messages qui ne sont pas envoyés en réponse directe à une commande MQSC.

Indiquez l'une des valeurs suivantes :

1. Valeur comprise entre 1 et 16 inclus.
2. Liste de valeurs, séparées par une virgule et entre parenthèses. Chaque valeur doit être comprise entre 1 et 16 inclus.

La valeur par défaut est 1.

Pour plus d'informations sur les codes de routage z/OS , voir *Codes de routage* dans [Description de message](#) dans l'un des volumes des manuels z/OS MVS System Messages .

SERVICE

Cette zone est réservée à l'utilisation par IBM.

SMFACCT

Indique si IBM MQ envoie automatiquement des données comptables à SMF lorsque le gestionnaire de files d'attente démarre.

Indiquez l'une des valeurs suivantes :

NO

Ne commencez pas à collecter automatiquement les données de comptabilité.

YES

Commencez à collecter automatiquement les données de comptabilité pour la classe par défaut 1.

entiers

Liste des classes pour lesquelles les données comptables sont collectées automatiquement entre 1 et 4.

* Démarrez automatiquement la comptabilité SMF pour les classes 1, 2 et 3.

La valeur par défaut est NO.

SMFSTAT

Indique si les statistiques SMF doivent être collectées automatiquement au démarrage du gestionnaire de files d'attente.

Indiquez l'une des valeurs suivantes :

NO

Ne commencez pas à collecter les statistiques automatiquement.

YES

Démarrer la collecte automatique des statistiques pour la classe par défaut 1.

entiers

Liste des classes pour lesquelles les statistiques sont collectées automatiquement dans la plage 1 à 5.

Pour collecter des statistiques de classe 2 ou 3, la classe 1 doit également être spécifiée.

* Démarrez les statistiques SMF automatiquement pour les classes 1, 2 et 3.

La valeur par défaut est NO.

SPLCAP

La fonction de stratégie de sécurité permet un niveau plus élevé de sécurité des messages via des stratégies qui contrôlent si les messages sont signés, ou chiffrés, lors de leur écriture et de leur lecture à partir des files d'attente.

Le traitement des stratégies de sécurité est configuré pour ce gestionnaire de files d'attente en définissant SPLCAP sur l'une des valeurs suivantes:

NO

La fonction d'implémentation des règles de sécurité des messages pour les files d'attente n'est pas activée lors de l'initialisation du gestionnaire de files d'attente.

YES

Les fonctions de sécurité des messages sont activées lors de l'initialisation du gestionnaire de files d'attente.

Le gestionnaire de files d'attente vérifie que l'attribut AMSPROD est défini sur AMS, ADVANCED ou ADVANCEDVUE, auquel cas il a été concédé sous licence pour AMS. Sinon, il ne démarrera pas.

Le gestionnaire de files d'attente vérifie également si la configuration AMS nécessaire est en place. Si ce n'est pas le cas, le gestionnaire de files d'attente ne démarrera pas.

Si le gestionnaire de files d'attente est sous licence pour AMS et que la configuration nécessaire est en place, le gestionnaire de files d'attente démarre avec les fonctions de sécurité des

messages activées lors de l'initialisation du gestionnaire de files d'attente et l'espace adresse AMSM est démarré.

La valeur par défaut est NO.

STATIME

Depuis la IBM MQ for z/OS 9.3.0, indique la durée, en minutes et en secondes, entre deux collectes consécutives de données statistiques. Si ACCTIME n'est pas défini ou est -1, indique également l'heure entre deux collectes consécutives de données comptables.

Indiquez un nombre compris entre 0 et 1440 minutes au format ' mmmm ' ou entre 0 et 1440 minutes et entre 0 et 59 secondes au format ' mmmm . ss '. La valeur par défaut est de 30 minutes.

Remarques :

- Lorsque vous spécifiez un intervalle de secondes uniquement, vous devez le préfixer avec la valeur 0. Le plus petit intervalle possible est d'une seconde: ' 0 . 01 '.
- Depuis la IBM MQ for z/OS 9.3.0, si vous spécifiez la valeur 0, les données statistiques sont collectées lors de la diffusion de la collecte de données SMF. Si ACCTIME n'est pas spécifié ou est -1, les données comptables sont également collectées lors de la diffusion de la collecte de données SMF. Pour plus d'informations, voir [Utilisation de System Management Facility](#) .
- Si vous spécifiez la valeur -1, qui est la valeur par défaut, les données comptables sont collectées à l'intervalle spécifié par la valeur STATIME.

TRACSTR

Indique si le traçage global doit démarrer automatiquement.

Indiquez l'une des valeurs suivantes :

NO

Ne démarrez pas la fonction de trace globale automatiquement.

YES

Démarrez automatiquement le traçage global pour la classe par défaut, la classe 1.

entiers

Liste des classes pour lesquelles la fonction de trace globale doit être démarrée automatiquement dans la plage 1 à 4.

Démarrer la trace globale automatiquement pour toutes les classes.

La valeur par défaut est NO si vous ne spécifiez pas le mot clé dans la macro.

Remarque : Le module de chargement des paramètres système par défaut fourni (CSQZPARM) a TRACSTR=YES (défini dans le module assembleur CSQFSYSP). Si vous ne souhaitez pas démarrer le traçage automatiquement, créez votre propre module de paramètres système ou émettez la commande STOP TRACE après le démarrage du gestionnaire de files d'attente.

Pour plus de détails sur la commande STOP TRACE, voir [STOP TRACE](#).

TRACTBL

Indique la taille par défaut, en blocs de 4 ko, de la table de trace dans laquelle la fonction de trace globale stocke les enregistrements de trace IBM MQ .

Indiquez une valeur comprise entre 1 et 999.

La valeur par défaut est 99. Cela équivaut à 396 Ko.

Remarque : La mémoire de la table de trace est allouée dans la zone de mémoire commune étendue (ECSA). Par conséquent, vous devez sélectionner cette valeur avec précaution.

WLMTIME

Indique la durée (en minutes ou secondes en fonction de la valeur de WLMTIMU) entre chaque analyse des index pour les files d'attente gérées par WLM.

Indiquez une valeur comprise entre 1 et 9999.

La valeur par défaut est 30.

WLMTIMU

Unités de temps utilisées avec le paramètre WLMTIME.

Indiquez l'une des valeurs suivantes:

minutes

WLMTIME représente un nombre de minutes.

SECS

WLMTIME représente un nombre de secondes.

La valeur par défaut est MINS.

Référence associée

«Using CSQ6LOGP», à la page 969

Use this topic as a reference for how to specify logging options using CSQ6LOGP.

«Using CSQ6ARVP», à la page 973

Use this topic as a reference for how to specify your archiving environment using CSQ6ARVP

Using CSQ6LOGP

Use this topic as a reference for how to specify logging options using CSQ6LOGP.

Use CSQ6LOGP to establish your logging options.

The default parameters for CSQ6LOGP, and whether you can alter each parameter using the SET LOG command, are shown in [Default values of CSQ6LOGP parameters](#). If you need to change any of these values, refer to the detailed descriptions of the parameters.

Parameter	Description	Default value	SET command
COMPLOG	Controls whether log compression is enabled.	NONE	X
DEALLCT	Length of time an archive tape unit remains unused before it is deallocated.	zero	X
INBUFF	Size of input buffer storage for active and archive log data sets.	60 KB	-
MAXARCH	Maximum number of archive log volumes that can be recorded.	500	X
MAXCNOFF	Maximum number of CSQJOFF7 offload tasks that can be run in parallel.	31	-
MAXRTU	Maximum number of dedicated tape units allocated to read archive log tape volumes concurrently.	2	X
OFFLOAD	Archiving on or off.	YES (ON)	-
OUTBUFF	Size of output buffer storage for active and archive log data sets.	4 000 KB	-
TWOACTV	Single or dual active logging.	YES (dual)	-
TWOARCH	Single or dual archive logging.	YES (dual)	-
TWOBSDS	Single or dual BSDS.	YES (dual BSDS)	-
WRTHRSH	Number of output buffers to be filled before they are written to the active log data sets.	20	X

Table 61. Default values of CSQ6LOGP parameters (continued)

Parameter	Description	Default value	SET command
ZHYWRITE	Specifies whether the zHyperWrite feature is enabled.	NO	X
V9.4.0 ZHYLINK	Specifies whether the zHyperLink feature is enabled.	NO	X

COMPLOG

Specifies whether log compression is enabled.

Specify either:

NONE

Log compression is not enabled.

RLE

Log compression is enabled using run-length encoding.

ANY

The queue manager selects the compression algorithm that gives the greatest degree of log record compression. This option results in RLE compression.

The default is NONE.

For more details about log compression, see [Log compression](#).

DEALLCT

Specifies the length of time, in minutes, that an archive read tape unit is allowed to remain unused before it is deallocated.

Specify one of the following:

- Time, in minutes, in the range zero through 1440
- NOLIMIT

Specifying 1440 or NOLIMIT means that the tape unit is never deallocated.

The default is zero.

When archive log data is being read from tape, it is recommended that you set this value high enough to allow IBM MQ to optimize tape handling for multiple read applications.

INBUFF

Specifies the size, in kilobytes, of the input buffer for reading the active and archive logs during recovery. Use a decimal number in the range 28 through 60. The value specified is rounded up to a multiple of 4.

The default is 60 KB.

Suggested settings:

Test system 28 KB

Production system 60 KB

Set this to the maximum for best log read performance.

MAXARCH

Specifies the maximum number of archive log volumes that can be recorded in the BSDS. When this number is exceeded, recording begins again at the start of the BSDS.

Use a decimal number in the range 10 through 1000.

The default is 500.

Suggested settings:

Test system 500 (default)

Production system 1 000

Set this to the maximum so that the BSDS can record as many logs as possible.

For information about the logs and BSDS, see [Managing IBM MQ resources](#).

MAXCNOFF

Specifies the number of CSQJOFF7 offload tasks that can be run in parallel.

This allows a queue manager, or queue managers, to be tuned such that they will not use all the available tape units.

Instead the queue manager waits until a CSQJOFF7 offload task has completed before trying to allocate any new archive data sets.

If the queue manager is archiving to tape, set this parameter so that the number of concurrent tape requests should not equal, or exceed, the number of tape units available, otherwise the system might hang.

Note that if dual archiving is in use, then each offload task performs both archives, so the parameter needs to be set accordingly. For example if the queue manager is dual archiving to tape, a value of MAXCNOFF=2 would allow up to two active logs to be archived concurrently to four tapes.

If several queue managers are sharing the tape units, you should set the MAXCNOFF for each queue manager accordingly.

The default value is 31.

Specify a value in the range 1 through 31.

MAXRTU

Specifies the maximum number of dedicated tape units that can be allocated to read archive log tape volumes concurrently.

This parameter and the DEALLCT parameter allow IBM MQ to optimize archive log reading from tape devices.

Specify a value in the range 1 through 99.

The default is 2.

It is recommended that you set the value to be at least one less than the number of tape units available to IBM MQ. If you do otherwise, the offload process could be delayed, which could affect the performance of your system. For maximum throughput during archive log processing, specify the largest value possible for this option, remembering that you need at least one tape unit for offload processing.

OFFLOAD

Specifies whether archiving is on or off.

Specify either:

YES

Archiving is on

NO

Archiving is off

The default is YES.

Attention: Do **not** switch archiving off unless you are working in a test environment. If you do switch it off, you cannot guarantee that data will be recovered in the event of a system or transaction failure.

OUTBUFF

Specifies the total size, in kilobytes, of the storage to be used by IBM MQ for output buffers for writing the active and archive log data sets. Each output buffer is 4 KB.

The parameter must be in the range 128 through 4000. The value specified is rounded up to a multiple of 4. Values between 40 and 128 will be accepted for compatibility reasons, and are treated as a value of 128.

The default is 4000 KB.

Suggested settings:

Test system	400 KB
Production system	4 000 KB

Set this value to the maximum to avoid running out of log output buffers.

TWOACTV

Specifies single or dual active logging.

Specify either:

NO

Single active logs

YES

Dual active logs

The default is YES.

For more information about the use of single and dual logging, see [Managing IBM MQ resources](#).

TWOARCH

Specifies the number of archive logs that IBM MQ produces when the active log is offloaded.

Specify either:

NO

Single archive logs

YES

Dual archive logs

The default is YES.

Suggested settings:

Test system	NO
Production system	YES (default)

For more information about the use of single and dual logging, see [Managing IBM MQ resources](#).

TWOBSDS

Specifies the number of bootstrap data sets.

Specify either:

NO

Single BSIDS

YES

Dual BSIDS

The default is YES.

For more information about the use of single and dual logging, see [Managing IBM MQ resources](#).

WRTHRSH

Specifies the number of 4 KB output buffers to be filled before they are written to the active log data sets.

The larger the number of buffers, the less often the write takes place, and this improves the performance of IBM MQ. The buffers might be written before this number is reached if significant events, such as a commit point, occur.

Specify the number of buffers in the range 1 through 256.

The default is 20.

ZHYWRITE

Indique si les écritures dans les journaux actifs sont effectuées alors que zHyperWrite est activé.

Pour plus d'informations, voir [Utilisation de zHyperWrite with IBM MQ active logs](#).

La valeur peut être :

NO

zHyperWrite n'est pas activé.

YES

zHyperWrite est activé.

V 9.4.0 ZHYLINK

Specifies whether writes to the active logs are made with zHyperLink being enabled.

For more information on enabling active logs with zHyperLink, see [Using zHyperLink with IBM MQ](#).

The value can be:

NO

zHyperLink is not enabled.

YES

zHyperLink is enabled.

Note: Enabling ZHYLINK also enables ZHYWRITE

Related reference

“Utilisation de CSQ6SYSP” on page 959

Utilisez cette rubrique comme référence pour définir les paramètres système à l'aide de CSQ6SYSP.

“Using CSQ6ARVP” on page 973

Use this topic as a reference for how to specify your archiving environment using CSQ6ARVP

z/OS Using CSQ6ARVP

Use this topic as a reference for how to specify your archiving environment using CSQ6ARVP

Use CSQ6ARVP to establish your archiving environment.

The default parameters for CSQ6ARVP, and whether you can alter each parameter using the SET ARCHIVE command, are shown in [Table 62 on page 973](#). If you need to change any of these values, refer to the detailed descriptions of the parameters. For more information about planning your storage, see [Planning your storage and performance requirements on z/OS](#).

Parameter	Description	Default value	SET command
ALCUNIT	Units in which primary and secondary space allocations are made.	BLK (blocks)	X
ARCPFX1	Prefix for first archive log data set name.	CSQARC1	X
ARCPFX2	Prefix for second archive log data set name.	CSQARC2	X

Table 62. Default values of CSQ6ARVP parameters (continued)

Parameter	Description	Default value	SET command
<u>ARCRETN</u>	The retention period of the archive log data set in days.	9999	X
<u>ARCWRTC</u>	List of route codes for messages to the operator about archive log data sets.	1,3,4	X
<u>ARCWTOR</u>	Whether to send message to operator and wait for reply before trying to mount an archive log data set.	YES	X
<u>BLKSIZE</u>	Block size of archive log data set.	28 672	X
<u>CATALOG</u>	Whether archive log data sets are cataloged in the ICF.	NO	X
<u>COMPACT</u>	Whether archive log data sets should be compacted.	NO	X
<u>PRIQTY</u>	Primary space allocation for DASD data sets.	25 715	X
<u>PROTECT</u>	Whether archive log data sets are protected by ESM profiles when the data sets are created.	NO	X
<u>QUIESCE</u>	Maximum time, in seconds, allowed for quiesce when ARCHIVE LOG with MODE(QUIESCE) specified.	5	X
<u>SECQTY</u>	Secondary space allocation for DASD data sets. See the ALCUNIT parameter for the units to be used.	540	X
<u>TSTAMP</u>	Whether the archive data set name should include a time stamp.	NO	X
<u>UNIT</u>	Device type or unit name on which the first copy of archive log data sets is stored.	TAPE	X
<u>UNIT2</u>	Device type or unit name on which the second copy of archive log data sets is stored.	Blank	X

ALCUNIT

Specifies the unit in which primary and secondary space allocations are made.

Specify one of:

CYL

Cylinders

TRK

Tracks

BLK

Blocks

You are recommended to use BLK because it is independent of the device type.

The default is BLK.

If free space on the archive DASD volumes is likely to be fragmented, you are recommended to specify a smaller primary extent and allow expansion into secondary extents. For more information about space allocation for active logs, refer to [Planning your log archive storage](#).

ARCPFX1

Specifies the prefix for the first archive log data set name.

See the TSTAMP parameter for a description of how the data sets are named and for restrictions on the length of ARCPFX1.

This parameter cannot be left blank.

The default is CSQARC1.

You might need to authorize the userid associated with the IBM MQ queue manager address space to create archive logs with this prefix.

ARCPFX2

Specifies the prefix for the second archive log data set name.

See the TSTAMP parameter for a description of how the data sets are named and for restrictions on the length of ARCPFX2.

This parameter cannot be blank even if the TWOARCH parameter is specified as NO.

The default is CSQARC2.

You might need to authorize the userid associated with the IBM MQ queue manager address space to create archive logs with this prefix.

ARCRETN

Specifies the retention period, in days, to be used when the archive log data set is created.

The parameter must be in the range zero through 9999.

The default is 9999.

Suggested settings:

Test system	3	In a test system, archive logs are probably not required over long periods.
Production system	9 999 (default)	Set this value high to effectively switch automatic archive log deletion off.

For more information about discarding archive log data sets, see [Discarding archive log data sets](#).

ARCWRTC

Specifies the list of z/OS routing codes for messages about the archive log data sets to the operator. This field is ignored if ARCWTOR is set to NO.

Specify up to 14 routing codes, each with a value in the range 1 through 16. You must specify at least one code. Separate codes in the list by commas, not by blanks.

The default is the list of values: 1,3,4.

For more information about z/OS routing codes, see *Routing codes* in [Message description](#) in one of the volumes of the *z/OS MVS System Messages* manuals.

ARCWTOR

Specifies whether a message is to be sent to the operator and a reply is received before attempting to mount an archive log data set.

Other IBM MQ users might be forced to wait until the data set is mounted, but they are not affected while IBM MQ is waiting for the reply to the message.

Specify either:

YES

The device needs a long time to mount archive log data sets. For example, a tape drive.

NO

The device does not have long delays. For example, DASD.

The default is YES.

Suggested settings:

Test system NO

Production system YES (default)

This is dependent on operational procedures. If tape robots are used, NO might be more appropriate.

BLKSIZE

Specifies the block size of the archive log data set. The block size you specify must be compatible with the device type you specify in the UNIT parameter.

The parameter must be in the range 4 097 through 28 672. The value you specify is rounded up to a multiple of 4 096.

The default is 28 672.

This parameter is overridden by the storage management subsystem (SMS) data class blocksize, if it is provided.

If the archive log data set is written to DASD, you are recommended to choose the maximum block size that allows two blocks for each track. For example, for a 3390 device, you should use a block size of 24 576.

If the archive log data set is written to tape, specifying the largest possible block size improves the speed of reading the archive log. You should use a block size of 28 672.

Suggested settings:

Test system Use the block size recommendation depending on the media used for archive logs.

That is, for disk 24 576, and tape 28 672.

Production system Use the block size recommendation depending on the media used for archive logs.

That is, for disk 24 576, and tape 28 672.

CATALOG

Specifies whether archive log data sets are cataloged in the primary integrated catalog facility (ICF) catalog.

Specify either:

NO

Archive log data sets are not cataloged

YES

Archive log data sets are cataloged

The default is NO.

All archive log data sets allocated on DASD must be cataloged. If you archive to DASD with the CATALOG parameter set to NO, message [CSQJ072E](#) is displayed each time an archive log data set is allocated, and IBM MQ catalogs the data set.

Suggested settings:

Test system YES

Production system YES, when archives are allocated on DASD

COMPACT

Specifies whether data written to archive logs is to be compacted. This option applies only to a 3480 or 3490 device that has the improved data recording capability (IDRC) feature. When this

feature is turned on, hardware in the tape control unit writes data at a much higher density than normal, allowing for more data on each volume. Specify NO if you do not use a 3480 device with the IDRC feature or a 3490 base model, except for the 3490E. Specify YES if you want the data to be compacted.

Specify either:

NO

Do not compact the data sets

YES

Compact the data sets

The default is NO.

Specifying YES adversely affects performance. Also be aware that data compressed to tape can be read only using a device that supports the IDRC feature. This can be a concern if you have to send archive tapes to another site for remote recovery.

Suggested settings:

Test system Not applicable

Production system NO (default)

This applies to 3480 and 3490 IDR compression only. Setting this to YES might degrade archive log read performance during recovery and restart; however, it does not affect writing to tape.

PRIQTY

Specifies the primary space allocation for DASD data sets in ALCUNITs.

The value must be greater than zero.

The default is 25 715.

This value must be sufficient for a copy of either the log data set or its corresponding BSDS, whichever is the larger. To determine the necessary value, follow this procedure:

1. Determine the number of active log records allocated (c) as explained in [“Create the bootstrap and log data sets” on page 955](#).
2. Determine the number of 4096 byte blocks in each archive log block:

$$d = \text{BLKSIZE} / 4096$$

where BLKSIZE is the rounded up value.

3. If ALCUNIT=BLK:

$$\text{PRIQTY} = \text{INT}(c / d) + 1$$

where INT means round down to an integer.

If ALCUNIT=TRK:

$$\text{PRIQTY} = \text{INT}(c / (d * \text{INT}(e/\text{BLKSIZE}))) + 1$$

where e is the number of bytes for each track (56664 for a 3390 device) and INT means round down to an integer.

If ALCUNIT=CYL:

$$\text{PRIQTY} = \text{INT}(c / (d * \text{INT}(e/\text{BLKSIZE}) * f)) + 1$$

where f is the number of tracks for each cylinder (15 for a 3390 device) and INT means round down to an integer.

For information about how large to make your log and archive data sets, see [“Create the bootstrap and log data sets”](#) on page 955 and [“Define your page sets”](#) on page 956.

Suggested settings:

Test system 1 680

Sufficient to hold the entire active log, that is:

$$10\ 080 / 6 = 1\ 680 \text{ blocks}$$

Production system Not applicable when archiving to tape.

If free space on the archive DASD volumes is likely to be fragmented, you are recommended to specify a smaller primary extent and allow expansion into secondary extents. For more information about space allocation for active logs, see [Planning your log archive storage](#).

PROTECT

Specifies whether archive log data sets are to be protected by discrete ESM (external security manager) profiles when the data sets are created.

Specify either:

NO

Profiles are not created.

YES

Discrete data set profiles are created when logs are offloaded. If you specify YES:

- ESM protection must be active for IBM MQ.
- The user ID associated with the IBM MQ queue manager address space must have authority to create these profiles.
- The TAPEVOL class must be active if you are archiving to tape.

Otherwise, offloading fails.

The default is NO.

QUIESCE

Specifies the maximum time in seconds allowed for the quiesce when an ARCHIVE LOG command is issued with MODE(QUIESCE) specified.

The parameter must be in the range 1 through 999.

The default is 5.

SECQTY

Specifies the secondary space allocation for DASD data sets in ALCUNITs. The secondary extent can be allocated up to 15 times; see the [IBM z/OS Management Facility Programming Guide](#) for more information on ALCUNIT.

The parameter must be greater than zero.

The default is 540.

TSTAMP

Specifies whether the archive log data set name has a time stamp in it.

Specify either:

NO

Names do not include a time stamp. The archive log data sets are named:

```
arcpfxi.A nnnnnnn
```

Where *arcpfxi* is the data set name prefix specified by ARCPFX1 or ARCPFX2. *arcpfxi* can have up to 35 characters.

YES

Names include a time stamp. The archive log data sets are named:

```
arcpfxi.cyyddd.T hhmsst.A nnnnnnn
```

where *c* is 'D' for the years up to and including 1999 or 'E' for the year 2000 and later, and *arcpfxi* is the data set name prefix specified by ARCPFX1 or ARCPFX2. *arcpfxi* can have up to 19 characters.

EXT

Names include a time stamp. The archive log data sets are named:

```
arcpfxi.D yyyddd.T hhmsst.A nnnnnnn
```

Where *arcpfxi* is the data set name prefix specified by ARCPFX1 or ARCPFX2. *arcpfxi* can have up to 17 characters.

The default is NO.

UNIT

Specifies the device type or unit name of the device that is used to store the first copy of the archive log data set.

Specify a device type or unit name of 1 through 8 alphanumeric characters. The first character must be alphabetic.

This parameter cannot be blank.

The default is TAPE.

If you archive to DASD, you can specify a generic device type with a limited volume range, for example, UNIT=3390.

If you archive to DASD, make sure that:

- The primary space allocation is large enough to contain all the data from the active log data sets.
- The archive log data set catalog option (CATALOG) is set to YES.
- You have used a proper value for BLKSIZE.

If you archive to TAPE, IBM MQ can extend to a maximum of 20 volumes.

Suggested settings:

Test system	DASD
Production system	TAPE

For more information about choosing a location for archive logs, see [Planning your log archive storage](#).

UNIT2

Specifies the device type or unit name of the device that is used to store the second copy of the archive log data sets.

Specify a device type or unit name of 1 through 8 alphanumeric characters. The first character must be alphabetic. If this parameter is blank, the value set for the UNIT parameter is used.

The default is blank.

Related reference

“Utilisation de CSQ6SYSP” on page 959

Utilisez cette rubrique comme référence pour définir les paramètres système à l'aide de CSQ6SYSP.

“Using CSQ6LOGP” on page 969

Use this topic as a reference for how to specify logging options using CSQ6LOGP.

Using CSQ6USGP

Use this topic as a reference for how to set your system parameters using CSQ6USGP

Use CSQ6USGP to control product usage recording.

The default parameters for CSQ6USGP are shown in [Table 63 on page 980](#). If you need to change any of these values, refer to the detailed descriptions of the parameters.



Attention: You cannot alter any of these parameters using the SET SYSTEM command.

Parameter	Description	Default value
QMGRPROD	Product against which queue manager usage is to be recorded	Blank
AMSPROD	Product against which Advanced Message Security (AMS) usage is to be recorded	Blank

QMGRPROD

Specifies the product against which queue manager usage is to be recorded.

Specify one of:

MQ

Queue manager usage is recorded as a stand-alone IBM MQ for z/OS product, with product ID 5655-MQ9.

VUE

Queue manager usage is recorded as a stand-alone IBM MQ for z/OS Value Unit Edition (VUE) product, with product ID 5655-VU9.

ADVANCEDVUE

Queue manager usage is recorded as part of an IBM MQ Advanced for z/OS Value Unit Edition product, with product ID 5655-AV1.

AMSPROD

If this parameter is not set the AMS address space will not start up and message [CSQY024I](#) will be output.

Specifies the product against which Advanced Message Security usage is to be recorded, if used.

Specify one of:

AMS

AMS usage is recorded as a stand-alone Advanced Message Security for z/OS product, with product ID 5655-AM9.

ADVANCED

AMS usage is recorded as part of an IBM MQ Advanced for z/OS product, with product ID 5655-AV9.

ADVANCEDVUE

AMS usage is recorded as part of an IBM MQ Advanced for z/OS Value Unit Edition product, with product ID 5655-AV1.

See [Reporting product information](#) for more information on product usage recording.

Related reference

[“Utilisation de CSQ6SYSP” on page 959](#)

Utilisez cette rubrique comme référence pour définir les paramètres système à l'aide de CSQ6SYSP.

[“Using CSQ6LOGP” on page 969](#)

Use this topic as a reference for how to specify logging options using CSQ6LOGP.

Tailor the channel initiator parameters

Use ALTER QMGR to customize the channel initiator to suit your requirements.

- Repeat this task for each IBM MQ queue manager, as required.
- You must perform this task when migrating from a previous version.

A number of queue manager attributes control how distributed queuing operates. Set these attributes using the MQSC command ALTER QMGR. The initialization data set sample thlqual.SCSQPROC(CSQ4INYG) contains some settings that you can customize. For more information, see [ALTER QMGR](#).

The values of these parameters are displayed as a series of messages each time you start the channel initiator.

The relationship between adapters, dispatchers, and maximum number of channels

The ALTER QMGR parameters CHIADAPS and CHIDISPS define the number of task control blocks (TCBs) used by the channel initiator. CHIADAPS (adapter) TCBs are used to make IBM MQ API calls to the queue manager. CHIDISPS (dispatcher) TCBs are used to make calls to the communications network.

The ALTER QMGR parameter MAXCHL influences the distribution of channels over the dispatcher TCBs.

CHIDISPS

If you have a small number of channels use the default value.

One task for each processor optimizes system performance. As dispatcher tasks are CPU intensive, the principle is to keep as few tasks as busy as possible, so that the time taken to find and start threads is minimized.

CHIDISPS(20) is suitable for systems with more than 100 channels. There is unlikely to be any significant disadvantage in having CHIDISPS(20) where this is more dispatcher TCBs than necessary.

As a guideline, if you have more than 1000 channels, allow one dispatcher for every 50 current channels. For example, specify CHIDISPS(40) to handle up to 2000 active channels.

If you are using TCP/IP, the maximum number of dispatchers used for TCP/IP channels is 100, even if you specify a larger value in CHIDISPS.

CHIADAPS

Each IBM MQ API call to the queue manager is independent of any other and can be made on any adapter TCB. Calls using persistent messages can take much longer than those for nonpersistent messages because of log I/O. Thus a channel initiator processing a large number of persistent messages across many channels may need more than the default 8 adapter TCBs for optimum performance. This is particularly so where achieved batchsize is small, because end of batch processing also requires log I/O, and where thin client channels are used.

The suggested value for a production environment is CHIADAPS(30). Using more than this is unlikely to give any significant extra benefit, and there is unlikely to be any significant disadvantage in having CHIADAPS(30) if this is more adapter TCBs than necessary.

MAXCHL

Each channel is associated with a particular dispatcher TCB at channel start and remains associated with that TCB until the channel stops. Many channels can share each TCB. MAXCHL is used to spread channels across the available dispatcher TCBs. The first ($\text{MIN}(\text{MAXCHL} / \text{CHIDISPS}), 10$) channels to start are associated with the first dispatcher TCB, and so on, until all dispatcher TCBs are in use.

The effect of this for small numbers of channels and a large MAXCHL is that channels are NOT evenly distributed across dispatchers. For example, if you set CHIDISPS(10) and left MAXCHL at its default value of 200 but had only 50 channels, five dispatchers would be associated with 10 channels each and five would be unused. We suggest setting MAXCHL to the number of channels actually to be used where this is a small fixed number.

If you change this queue manager property, you must also review the ACTCHL, LU62CHL, and TCPCHL queue manager properties to ensure that the values are compatible. See [Queue manager parameters](#) for a full description of these properties, and their relationship.

Setting up your z/OS UNIX System Services environment for channel initiators

The channel initiator (CHINIT) uses OMVS threads. Review the OMVS configuration parameters before creating a new CHINIT, or modifying the number of dispatchers or SSLTASKS.

Each CHINIT uses $3 + \text{CHIDISP} + \text{SSLTASKS}$ OMVS threads. These contribute to the total number of OMVS threads used in the LPAR, and towards the number of threads used by CHINIT started task user ID.

You can use the **D OMVS,L** and review the current usage, highwater usage, and system limit of MAXPROCSYS (the maximum number of processes that the system allows).

If you are adding a new CHINIT or increasing the values of CHIDISPS or SSLTASKS then you must calculate the increase in threads and review the impact on the MAXPROCSYS values. You can use the **SETOMVS** command to dynamically change the MAXPROCSYS, or update the BPXPRCxx parmlib value or both.

The OMVS parameter MAXPROCUSER is the number of OMVS threads a single OMVS user, that is with the same UID, can have. The threads count towards this value. So if you have 2 CHINITs with the same started task user ID, with 10 dispatchers and 3 SSLTASKS each then there are $2 * (3 + 10 + 3) = 32$ threads for the OMVS uid.

You can display the default MAXPROCUSER by issuing the **D OMVS,O** command and you can use the **SETOMVS** command to dynamically change the MAXPROCUSER, or update the BPXPRCxx parmlib value or both.

You can override this value on a per user basis with the RACF command **ALTUSER userid OMVS (PROCUSERMAX(nnnn))** or equivalent.

To start the channel initiator, issue the following command:

```
START CHINIT
```

To ensure that the channel initiator has started successfully, check that there is no ICH408I error in the xxxxCHIN(ssidCHIN) job log.

Related concepts

[“Set up Batch, TSO, and RRS adapters” on page 983](#)

Make the adapters available to applications by adding libraries to appropriate STEPLIB concatenations. To cater for SNAP dumps issued by an adapter, allocate a CSQSNAP DDname. Consider using CSQBDEFV to improve the portability of your application programs

Related reference

[Channel initiator statistics data records](#)

Set up Batch, TSO, and RRS adapters

Make the adapters available to applications by adding libraries to appropriate STEPLIB concatenations. To cater for SNAP dumps issued by an adapter, allocate a CSQSNAP DDname. Consider using CSQBDEFV to improve the portability of your application programs

- *Repeat this task for each IBM MQ queue manager as required.*
- *You might need to perform this task when migrating from a previous version.*

To make the adapters available to batch and other applications using batch connections, add the following IBM MQ libraries to the STEPLIB concatenation for your batch application :

- thlqual.SCSQANL x
- thlqual.SCSQAUTH

where x is the language letter for your national language. (You do not need to do this if the libraries are in the LPA or the link list.)

For TSO applications add the libraries to the STEPLIB concatenation in the TSO logon procedure or activate them using the TSO command TSOLIB.

If the adapter detects an unexpected IBM MQ error, it issues an z/OS SNAP dump to DDname CSQSNAP, and issues reason code MQRC_UNEXPECTED_ERROR to the application. If the CSQSNAP DD statement is not in the application JCL or CSQSNAP is not allocated to a data set under TSO, no dump is taken. If this happens, you could include the CSQSNAP DD statement in the application JCL or allocate CSQSNAP to a data set under TSO and rerun the application. However, because some problems are intermittent, it is recommended that you include a CSQSNAP statement in the application JCL or allocate CSQSNAP to a data set in the TSO logon procedure to capture the reason for failure at the time it occurs.

The supplied program CSQBDEFV improves the portability of your application programs. In CSQBDEFV, you can specify the name of a queue manager, or queue sharing group, to be connected to rather than specifying it in the MQCONN or MQCONNx call in an application program. You can create a new version of CSQBDEFV for each queue manager, or queue sharing group. To do this, follow these steps:

1. Copy the IBM MQ assembler program CSQBDEFV from thlqual.SCSQASMS to a user library.
2. The supplied program contains the default subsystem name CSQ1. You can retain this name for testing and installation verification. For production subsystems, you can change the NAME=CSQ1 to your one-to four-character subsystem name, or use CSQ1.

If you are using queue sharing groups, you can specify a queue sharing group name instead of CSQ1. If you do this, the program issues a connect request to an active queue manager within that group.

3. Assemble and link-edit the program to produce the CSQBDEFV load module. For the assembly, include the library thlqual.SCSQMACS in your SYSLIB concatenation; use the link-edit parameters RENT, AMODE=31, RMODE=ANY. This is shown in the sample JCL in thlqual.SCSQPROC(CSQ4DEFV). Then include the load library in the z/OS Batch or the TSO STEPLIB, ahead of thlqual.SCSQAUTH.

Related concepts

[“Set up the operations and control panels” on page 983](#)

To set up the operations and control panels you must first set up the libraries that contain the required panels, EXECs, messages, and tables. To do this, you must take into account which national language feature is to be used for the panels. When you have done this, you can optionally update the main ISPF menu for IBM MQ operations and control panels and change the function key settings.

Set up the operations and control panels

To set up the operations and control panels you must first set up the libraries that contain the required panels, EXECs, messages, and tables. To do this, you must take into account which national language feature is to be used for the panels. When you have done this, you can optionally update the main ISPF menu for IBM MQ operations and control panels and change the function key settings.

- You need to perform this task once for each z/OS system where you want to run IBM MQ.
- You might need to perform this task when migrating from a previous version.

Setting up the libraries

Follow these steps to set up the IBM MQ operations and control panels:

1. Ensure that all the libraries contained in your concatenations are either in the same format (F, FB, V, VB) and have the same block size, or are in order of decreasing block sizes. Otherwise, you might have problems trying to use these panels.
2. Include the library thlqual.SCSQEXEC in your SYSEXEC or SYSPROC concatenation or activate it using the TSO ATLIB command. This library, which is allocated with a fixed-block 80 record format during installation, contains the required EXECs.

It is preferable to put the library into your SYSEXEC concatenation. However, if you want to put it in SYSPROC, the library must have a record length of 80 bytes.

3. Add thlqual.SCSQAUTH and thlqual.SCSQANLx to the TSO logon procedure STEPLIB or activate it using the TSO TSOLIB command, if it is not in the link list or the LPA.
4. You can either add the IBM MQ panel libraries permanently to your ISPF library setup, or allow them to be set up dynamically when the panels are used. For the former choice, you need to do the following:
 - a. Include the library containing the operations and control panel definitions in your ISPLIB concatenation. The name is thlqual.SCSQPNLx, where x is the language letter for your national language.
 - b. Include the library containing the required tables in your ISPTLIB concatenation. The name is thlqual.SCSQTBLx, where x is the language letter for your national language.
 - c. Include the library containing the required messages in your ISPMLIB concatenation. The name is thlqual.SCSQMSGx, where x is the language letter for your national language.
 - d. Include the library containing the required load modules in your ISPLLIB concatenation. The name of this library is thlqual.SCSQAUTH.

For the latter choice, use the z/OS [LIBDEF](#) command. See [Examples](#) for a link to various keywords you can use.

5. Test that you can access the IBM MQ panels from the TSO Command Processor panel. This is usually option 6 on the ISPF/PDF Primary Options Menu. The name of the EXEC that you run is CSQOREXX. There are no parameters to specify if you have put the IBM MQ libraries permanently in your ISPF setup as in step 4. If you have not, use the following:

```
CSQOREXX thlqual langletter
```

where langletter is a letter identifying the national language to be used:

- C** Simplified Chinese
- E** U.S. English (mixed case)
- F** French
- K** Japanese
- U** U.S. English (uppercase)

Updating the ISPF menu

You can update the ISPF main menu to allow access to the IBM MQ operations and control panels from ISPF. The required setting for &ZSEL is:

```
CMD(%CSQOREXX thlqual langletter)
```

For information about thlqual and langletter, see Step “5” on page 984.

For more details, see the [z/OS: ISPF Dialog Developer's Guide and Reference](#).

Updating the function keys and command settings

You can use the normal ISPF procedures for changing the function keys and command settings used by the panels. The application identifier is CSQO.

However, this is not recommended because the help information is not updated to reflect any changes that you have made.

Related concepts

[“Include the IBM MQ dump formatting member” on page 985](#)

To be able to format IBM MQ dumps using the Interactive Problem Control System (IPCS), you must update some system libraries.

Include the IBM MQ dump formatting member

To be able to format IBM MQ dumps using the Interactive Problem Control System (IPCS), you must update some system libraries.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *You need to perform this task when migrating from a previous version.*

To be able to format IBM MQ dumps using the Interactive Problem Control System (IPCS), copy the data set thlqual.SCSQPROC(CSQ7IPCS) to SYS1.PARMLIB. You should not need to edit this data set.

If you have customized the TSO procedure for IPCS, thlqual.SCSQPROC(CSQ7IPCS) can be copied into any library in the IPCSPARM definition. See [z/OS MVS IPCS User's Guide](#) for more information.

You must also include the library thlqual.SCSQPnLA in your ISPPnLIB concatenation.

To make the dump formatting programs available to your TSO session or IPCS job, you must also include the library thlqual.SCSQAUTH in your STEPLIB concatenation or activate it using the TSO TSOLIB command (even if it is already in the link list or LPA).

Related concepts

[“Suppress information messages” on page 985](#)

Your IBM MQ system might produce a large number of information messages. You can prevent selected messages being sent to the console or to the hardcopy log.

Suppress information messages

Your IBM MQ system might produce a large number of information messages. You can prevent selected messages being sent to the console or to the hardcopy log.

- *You need to perform this task once for each z/OS system where you want to run IBM MQ.*
- *You do not need to perform this task when migrating from a previous version.*

If your IBM MQ system is heavily used, with many channels stopping and starting, a large number of information messages are sent to the z/OS console and hardcopy log. The IBM MQ - IMS bridge and buffer manager might also produce a large number of information messages.

If required, you can suppress some of these console messages by using the z/OS message processing facility list, specified by the MPFLSTxx members of SYS1.PARMLIB. The messages you specify still appear on the hardcopy log, but not on the console.

Sample th1qua1.SCSQPROC(CSQ4MPFL) shows suggested settings for MPFLSTxx. See [MPFLSTxx \(message processing facility list\)](#) for more information.

If you want to suppress selected information messages on the hardcopy log, you can use the z/OS installation exit IEAVMXIT. You can set the following bit switches ON for the required messages:

CTXTRDTM

Delete the message.

The message is not displayed on consoles or logged in hardcopy.

CTXTESJL

Suppress from job log.

The message does not go into the JES job log.

CTXTNWTP

Do not carry out WTP processing.

The message is not sent to a TSO terminal or to the system message data set of a batch job.

Note:

1. For full details on the other parameters, see [MVS Installation Exits](#).
2. You are not recommended to suppress messages other than those in the suggested suppression list, CSQ4MPFL.

In addition you can specify the extra parameter:

EXCLMSG

Specifies a list of messages to be excluded from any log.

Messages in this list are not sent to the z/OS console and hardcopy log. See [EXCLMSG](#) in [“Utilisation de CSQ6SYSP”](#) on page 959 for further information.

Related tasks

[“Testing a queue manager on z/OS”](#) on page 1001

When you have customized or migrated your queue manager, you can test it by running the installation verification programs and some of the sample applications shipped with IBM MQ for z/OS.

Configuring the queue sharing group

If you want to use shared queues for high availability, use these topics as a step by step guide for configuring the queue sharing group.

When you have completed the steps in this part of the process for setting up your IBM MQ for z/OS system, you should [“Tailor your system parameter module”](#) on page 957 to add queue sharing group data. You need to modify [CSQ6SYSP](#) to specify the QSGDATA parameter.

Set up the Db2 environment

If you are using queue sharing groups you must create the required Db2 objects by customizing and running a number of sample jobs.

Set up the Db2 environment

You must create and bind the required Db2 objects by customizing and running a number of sample jobs.

- Repeat this task for each Db2 data-sharing group.
- You need to perform the bind and grant steps when migrating from a previous version.
- Omit this task if you are not using queue sharing groups.

If you later want to use queue sharing groups, perform this task at that time.

IBM MQ provides two equivalent sets of jobs. Those with the CSQ45 prefix are for compatibility with earlier versions of IBM MQ and for use with IBM MQ version 11 and earlier. If you are setting up a new data-sharing group with Db2 V12 or later, you are encouraged to use the jobs with CSQ4X prefix, as these jobs exploit more recent Db2 capabilities for dynamic sizing and Universal Table Spaces (UTS).

The following steps must be performed for each new Db2 data-sharing group. All the sample JCL is in thlqual.SCSQPROC.

1. Customize and execute sample JCL CSQ4XCSG to create the storage group that is to be used for the IBM MQ database, table spaces, and tables.
2. Customize and execute sample JCL CSQ4XCDB to create the database to be used by all queue managers that are connecting to this Db2 data-sharing group.
3. Customize and execute sample JCL CSQ4XCTS to create the table spaces that contain the queue manager and channel initiator tables used for queue sharing groups.
4. Customize and execute sample JCL CSQ4XCTB to create the 15 Db2 tables and associated indexes. Do not change any of the row names or attributes.
5. Customize and execute sample JCL CSQ45BPL to bind the Db2 plans for the queue manager, utilities, and channel initiator.
6. Customize and execute sample JCL CSQ45GEX to grant execute authority to the plans for the user IDs that are used by the queue manager, utilities, and channel initiator. The user IDs for the queue manager and channel initiator are the user IDs under which their started task procedures run. The user IDs for the utilities are the user IDs under which the batch jobs can be submitted.

The names of the appropriate plans are shown in the following table.

User	Plans (LTS)	Plans (CD)
Queue manager	CSQ5A 930, CSQ5C 930, CSQ5D 930, CSQ5K 930, CSQ5L 930, CSQ5M 930, CSQ5P 930, CSQ5R 930, CSQ5S 930, CSQ5T 930, CSQ5U 930, CSQ5W 930	CSQ5A 9X0, CSQ5C 9X0, CSQ5D 9X0, CSQ5K 9X0, CSQ5L 9X0, CSQ5M 9X0, CSQ5P 9X0, CSQ5R 9X0, CSQ5S 9X0, CSQ5T 9X0, CSQ5U 9X0, CSQ5W 9X0
SDEFS function of the CSQUTIL batch utility	CSQ52 930	CSQ52 9X0
CSQ5PQSG and CSQJUCNV batch utilities	CSQ5B 930	CSQ5B 9X0
CSQUZAP service utility	CSQ5Z 930	CSQ5Z 9X0

In the event of a failure during Db2 setup, the following jobs can be customized and executed:

- CSQ45DTB to drop the tables and indexes.
- CSQ4XDTS to drop the table spaces.
- CSQ4XDDB to drop the database.
- CSQ4XDSG to drop the storage group.

Note: If these jobs fail because of a Db2 locking problem it is probably due to contention for a Db2 resource, especially if the system is being heavily used. Resubmit the jobs later. It is preferable to run these jobs when the system is lightly used or quiesced.

See [Db2 Administration](#) in *Db2 for z/OS 12.0.0* for more information about setting up Db2.

See [Planification sur z/OS](#) for information about Db2 table sizes.

Related concepts

[“Set up the coupling facility” on page 988](#)

If you are using queue sharing groups, define the coupling facility structures used by the queue managers in the queue sharing group (QSG) in the coupling facility Resource Management (CFRM) policy data set, using IXCMIAPU.

Set up the coupling facility

If you are using queue sharing groups, define the coupling facility structures used by the queue managers in the queue sharing group (QSG) in the coupling facility Resource Management (CFRM) policy data set, using IXCMIAPU.

See [Administrative data utility](#) for more information on IXCMIAPU.

- Repeat this task for each queue sharing group.
- You might need to perform this task when migrating from a previous version.
- Omit this task if you are not using queue sharing groups.

If you later want to use queue sharing groups, perform this task at that time.

All the structures for the queue sharing group start with the name of the queue sharing group. Define the following structures:

- An administrative structure called *qsg-name* CSQ_ADMIN. This structure is used by IBM MQ itself and does not contain any user data.
- A system application structure called *qsg-name* CSQSYSAPPL. This structure is used by IBM MQ system queues to store state information.
- One or more structures used to hold messages for shared queues. These can have any name you choose up to 16 characters long.
 - The first four characters must be the queue sharing group name. (If the queue sharing group name is less than four characters long, it must be padded to four characters with @ symbols.)
 - The fifth character must be alphabetic and subsequent characters can be alphabetic or numeric. This part of the name (without the queue sharing group name) is what you specify for the CFSTRUCT name when you define a shared queue, or a CF structure object.

You can use only alphabetic and numeric characters in the names of structures used to hold messages for shared queues, you cannot use any other characters (for example, the _ character, which is used in the name of the administrative structure).

Sample control statements for IXCMIAPU are in data set thlqual.SCSQPROC(CSQ4CFRM). Customize these and add them to your IXCMIAPU job for the coupling facility and run it.

When you have defined your structures successfully, activate the CFRM policy that is being used. To do this, issue the following z/OS command:

```
SETXCF START,POLICY,TYPE=CFRM,POLNAME= policy-name
```

For information about planning CF structures and their sizes, see [Defining coupling facility resources](#).

Related concepts

[“Implement your ESM security controls” on page 949](#)

Implement security controls for queue managers and the channel initiator.

Set up the SMDS environment

If you want to use SMDS to offload messages on shared queues, set up the SMDS offload storage environment.

- *Perform this task for each queue manager and structure in the queue sharing group that you want to configure to offload data to SMDS.*
- *If you want to configure additional structures to offload data to SMDS later, this task can be performed again at that time.*
- *Omit this task if you are not using queue sharing groups.*

If you later want to use queue sharing groups, perform this task at that time.

Set up the SMDS environment

1. Estimate structure and data set space requirements. See [Shared message data set capacity considerations](#).
2. Allocate and preformat data sets. See [Creating a shared message data set](#).
3. When you define the CF structure to IBM MQ, ensure that you define the CFSTRUCT with CFLEVEL(5) and OFFLOAD(SMDS).

Related concepts

[“Set up the coupling facility” on page 988](#)

If you are using queue sharing groups, define the coupling facility structures used by the queue managers in the queue sharing group (QSG) in the coupling facility Resource Management (CFRM) policy data set, using IXCMIAPU.

Add the IBM MQ entries to the Db2 tables

If you are using queue sharing groups, run the CSQ5PQSG utility to add queue sharing group and queue manager entries to the IBM MQ tables in the Db2 data-sharing group.

- *Repeat this task for each IBM MQ queue sharing group and each queue manager.*
- *You might need to perform this task when migrating from a previous version.*
- *Omit this task if you are not using queue sharing groups.*

If you later want to use queue sharing groups, perform this task at that time.

Run [CSQ5PQSG](#) for each queue sharing group and each queue manager that is to be a member of a queue sharing group.

Perform the following actions in the specified order:

1. Add a queue sharing group entry into the IBM MQ Db2 tables using the ADD QSG function of the CSQ5PQSG program. A sample is provided in thlqual.SCSQPROC(CSQ45AQS).

Perform this function once for each queue sharing group that is defined in the Db2 data-sharing group. The queue sharing group entry must exist before adding any queue manager entries that reference the queue sharing group.

2. Add a queue manager entry into the IBM MQ Db2 tables using the ADD QMGR function of the CSQ5PQSG program. A sample is provided in thlqual.SCSQPROC(CSQ45AQM).

Perform this function for each queue manager that is to be a member of the queue sharing group.

Note:

- a. A queue manager can only be a member of one queue sharing group.
- b. You must have RRS running to be able to use queue sharing groups.

Related concepts

[“Tailor your system parameter module” on page 957](#)

The IBM MQ system parameter module controls the logging, archiving, tracing, and connection environments that IBM MQ uses in its operation. A default module is supplied. You should create your own system parameter module as some parameters, for example data set names, are usually site specific.

Implement ESM security controls for the queue sharing group

Implement security controls for all queue managers in a queue sharing group, to access Db2 and the coupling facility list structures.

- Repeat this task for each IBM MQ queue manager in a queue sharing group.
- You might need to perform this task when migrating from a previous version.

Ensure that the user IDs associated with the queue manager, channel initiator, and the utilities have authority to establish an RRSF connection to each Db2 subsystem with which you want to establish a connection. The user IDs for the queue manager and channel initiator are the user IDs under which their started task procedures run.

The user IDs for the utilities are the user IDs under which the batch jobs can be submitted. The RACF profile to which the user ID requires READ access is Db2ssid . RRSF in the DSNR resource class

The user IDs associated with each queue manager in a queue sharing group need to be granted the appropriate level of access to the coupling facility list structures. The RACF class is FACILITY.

The following user IDs require ALTER access:

- The queue manager ID to the IXLSTR . structure - name profile
- The user ID running CSQ5PQSG

Related concepts

[“Implement your ESM security controls” on page 949](#)

[Implement security controls for queue managers and the channel initiator.](#)

Configuring Advanced Message Security for z/OS

Use these topics as a step by step guide for configuring Advanced Message Security (AMS).

Before you begin

Before you start to configure AMS, ensure that the following queue manager configuration steps have been performed:

1. Add the CSQ0DRTM module to the LPA, as described in [“Mise à jour de la liste de liens z/OS et LPA” on page 938.](#)
2. Add an entry for CSQ0DSRV to the z/OS program properties table (PPT), as described in [“Update the z/OS program properties table” on page 942.](#)
3. Include the CSQ4INSM member in the CSQINP2 concatenation of queue manager started task procedure, as described in [“Customize the initialization input data sets” on page 950.](#)
4. Enable AMS using the AMSPROD attribute. See [product usage recording with IBM MQ for z/OS products](#) for more details.

What to do next

Configure policies for queues protected by AMS. Security policies are described in [Administering Advanced Message Security security policies.](#)

There are examples of AMS configurations in [Example configurations on z/OS.](#)

Create procedures for Advanced Message Security

Each IBM MQ subsystem that is to be configured to use Advanced Message Security (AMS) requires a cataloged procedure to start the AMS address space. You can create your own or use the IBM-supplied procedure library.

Procedure

1. Copy the sample started task procedure `thlqual.SCSQPROC(CSQ4AMSM)` to your `SYS1.PROCLIB` or, if you are not using `SYS1.PROCLIB`, your procedure library. Name the procedure `xxxxAMSM`, where `xxxx` is the name of your IBM MQ subsystem. For example, `CSQ1AMSM` would be the AMS started task procedure for queue manager `CSQ1`.
2. Make a copy for each IBM MQ subsystem that you are going to use.
3. Tailor the procedures to your requirements using the instructions in the sample procedure `CSQ4AMSM`. You can also use symbolic parameters in the JCL to allow the procedure to be modified when it is started.
4. Review and optionally change the parameters passed to the AMS task using the Language Environment® `_CEE_ENVFILE` file. The sample `thlqual.SCSQPROC(CSQ40ENV)` lists the supported parameters.
5. Repeat steps 1 to 4 for each IBM MQ queue manager.

What to do next

[“Set up the Advanced Message Security started task user ID” on page 991](#)

Set up the Advanced Message Security started task user ID

The Advanced Message Security (AMS) task requires a user ID that allows it to be known as a z/OS UNIX System Services (z/OS UNIX) process.

About this task

In addition, the users that the task works on behalf of must also have an appropriate definition of a UNIX UID (user ID) and GID (group ID) so these users are known as z/OS UNIX System Services users. For more information on defining z/OS UNIX System Services UIDs and GIDs, see [z/OS: Security Server RACF Security Administrator's Guide](#).

Review [z/OS UNIX System Services Planning](#) to ensure that you understand the security differences between traditional UNIX security and z/OS UNIX security. This allows you to administer the Advanced Message Security task according to your installation's security policy for deploying and running privileged z/OS UNIX System Services processes.

The primary difference between traditional UNIX security and z/OS security is that the Kernel services support two levels of appropriate privileges: UNIX level and z/OS UNIX level.

Depending on your installation's security policy, the Advanced Message Security task can either run with superuser authority (`uid(0)`), or with its RACF identity permitted to the RACF FACILITY class `BPX.DAEMON` and `BPX.SERVER` profiles, as this task must be able to assume the RACF identity of its users.

If the latter method is used, or you have already activated the `BPX.DAEMON` or `BPX.SERVER` profiles, the Advanced Message Security task program (`thlqual.SCSQAUTH(CSQ0DSRV)`) must be located in RACF program-controlled libraries.

Note: Choose the user ID for this task carefully because the Advanced Message Security recipient certificates are loaded into a key ring associated with this user ID. This consideration is discussed in [Using certificates on z/OS](#).

The steps shown here describe how to set up the Advanced Message Security started task user. The steps use RACF commands as examples. If you are using a different security manager, you should use equivalent commands.

Note: The examples in this section assume that you have activated generic profile command processing for the RACF STARTED, FACILITY, and SURROGAT classes and generic profile checking. For more information on how RACF handles generic profiles, see [z/OS: Security Server RACF Command Language Reference](#).

Procedure

1. Define the Advanced Message Security started task user to RACF. The examples in this section use the user ID WMQAMSM.

```
ADDUSER WMQAMSM NAME('AMS user') OMVS (UID(0)) DFLTGRP(group)
```

Select a default 'group' as appropriate to your installation standards.

Note: If you do not want to grant z/OS UNIX superuser authority (UID(0)), then you must permit the Advanced Message Security user ID to the BPX.DAEMON and BPX.SERVER facility class profiles:

```
PERMIT BPX.DAEMON CLASS(FACILITY) ID(WMQAMSM) ACCESS(READ)
```

and the Advanced Message Security task program (*thlqual.SCSQAUTH(CSQ0DSRV)*) must be located in a RACF program-controlled library.

To make your SCSQAUTH library program controlled, you can use the following command:

```
RALTER PROGRAM * ADDMEM('thlqual.SCSQAUTH'//NOPADCHK) -or-  
RALTER PROGRAM ** ADDMEM('thlqual.SCSQAUTH'//NOPADCHK)  
SETROPTS WHEN(PROGRAM) REFRESH
```

You must also enable program control for the national language library (*thlqual.SCSQANLx*) that is used by the Advanced Message Security task.

2. Determine if the RACF STARTED class is active. If it is not, activate the RACF STARTED class:

```
SETROPTS CLASSACT(STARTED)
```

3. Define a started class profile for the Advanced Message Security tasks, specifying the user ID you selected or created in step 1:

```
RDEFINE STARTED qmgrAMSM.* STDATA(USER(WMQAMSM))
```

where *qmgr* is the prefix of the started task name. For example, the started task may be named CSQ1AMSM. In this case, you would substitute *qmgr*AMSM.* with CSQ1AMSM.*.

The AMS started tasks must be named *qmgr*AMSM.

4. Use the **SETROPTS** RACF command to refresh the in-storage RACLISTed STARTED class profiles:

```
SETROPTS RACLIST(STARTED) REFRESH
```

5. The Advanced Message Security task temporarily assumes the identity of the host user ID of the requestor during protection processing of IBM MQ messages. Therefore, it is necessary to define profiles in the SURROGAT class for each user ID that can make requests.

If the RACF SURROGAT class is active, defining a single generic profile allows the Advanced Message Security task to assume the identity of any user. The check is ignored if the SURROGAT class is not active. The SURROGAT profiles needed are described in [z/OS UNIX System Services Planning](#).

To define profiles in the SURROGAT class:

- a) Activate the RACF SURROGAT class using the RACF SETROPTS command:


```
SETROPTS CLASSACT(SURROGAT)
```

- b) Activate generic profile processing for the RACF SURROGAT class:

```
SETROPTS GENERIC(SURROGAT)
```

- c) Activate generic profile command processing for the RACF SURROGAT class:

```
SETROPTS GENCMD(SURROGAT)
```

- d) Define a generic profile in the SURROGAT class:

```
RDEFINE SURROGAT BPX.SRV.* UACC(NONE)
```

- e) Permit the Advanced Message Security user ID to the generic SURROGAT class profile:

```
PERMIT BPX.SRV.* CLASS(SURROGAT) ID(WMQAMSM) ACCESS(READ)
```

Note: You can define more specific profiles if you want to restrict specific users to be processed by the Advanced Message Security task, as described in [z/OS UNIX System Services Planning](#).

For example, a profile called BPX.SRV.MQUSER1 controls whether the AMS task can assume the identity of the user ID MQUSER1.

- f) Permit the Advanced Message Security user ID to the BPX.SERVER facility (if not already done in [Creating the certificates and key rings](#)):

```
PERMIT BPX.SERVER CLASS(FACILITY) ID(WMQAMSM) ACCESS(READ)
```

- g) Use the **SETROPTS** RACF command to refresh the in-storage RACLISTed started class profiles:

```
SETROPTS RACLIST(SURROGAT) REFRESH  
SETROPTS RACLIST(FACILITY) REFRESH
```

6. The Advanced Message Security task uses the facilities provided by z/OS System SSL services to open SAF-managed key rings. The underlying System Authorization Facility (SAF) that accesses the contents of the key rings is controlled by RACF, or an equivalent security manager.

This service is the IRRSDL00 (R_datalib) callable service. This callable service is protected with the same profiles used to protect the RACF RACDCERT commands that are defined to the RACF FACILITY class. Thus, the Advanced Message Security user ID must be permitted to the profiles using these commands:

- a) If you have not already done so, define a RACF generic profile to the RACF FACILITY class that protects the RACDCERT command and the IRRSDL00 callable service:

```
RDEFINE FACILITY IRR.DIGTCERT.* UACC(NONE)  
SETROPTS RACLIST(FACILITY) REFRESH
```

- b) Grant authority to the started task user ID to the RACF generic profile:

```
PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID(WMQAMSM) ACC(READ)
```

Alternatively, you can grant READ access to the data service task user's keyring in the RDATA LIB class as follows:

```
PERMIT WMQASMD.DRQ.AMS.KEYRING.LST CLASS(RDATA LIB) ID(WMQAMSM) ACC(READ)
```

7. Configure resource security:

- a) The Advanced Message Security started task user requires authority to connect to the queue manager as a batch application.

If your queue manager has connection security enabled, grant the AMS task authority to connect to the queue manager with this command:

```
PERMIT h1q.BATCH CLASS(MQCONN) ID(WMQAMSM) ACC(READ)
```

where *h1q* can be either the queue manager name queue sharing group name.

For further information, see [Connection security profiles for batch connections](#).

- b) The Advanced Message Security started task user requires authority to browse the SYSTEM.PROTECTION.POLICY.QUEUE.

If queue security is active on the queue manager, grant the AMS user authority to access the queue with these commands:

```
RDEFINE MQQUEUE h1q.SYSTEM.PROTECTION.POLICY.QUEUE UACC(NONE)  
PERMIT h1q.SYSTEM.PROTECTION.POLICY.QUEUE CLASS(MQQUEUE) ID(WMQAMSM) ACCESS(READ)
```

where *h1q* can be either the queue manager name queue sharing group name.

If the queue manager is using mixed case profiles, define the profile in the MXQUEUE class instead.

To manage AMS security policies using the CSQOUTIL utility, administrators need access to put messages to the SYSTEM.PROTECTION.POLICY.QUEUE. This is performed by granting UPDATE access to the profile protecting the queue.

For further information, see [Profiles for queue security](#).

What to do next

[“Grant RACDCERT permissions to the security administrator for Advanced Message Security” on page 994](#)

Grant RACDCERT permissions to the security administrator for Advanced Message Security

Your Advanced Message Security security administrator requires authority to use the RACDCERT command to create and manage digital certificates.

Procedure

- Identify the appropriate user ID for this role and grant permission to use the RACDCERT command. For example:

```
PERMIT IRR.DIGTCERT.* CLASS(FACILITY) ID(admin) ACCESS(CONTROL)  
SETROPTS RACLIST(FACILITY) REFRESH
```

where *admin* is the user ID of your Advanced Message Security security administrator.

What to do next

[“Grant users resource permissions for Advanced Message Security” on page 994](#)

Grant users resource permissions for Advanced Message Security

Advanced Message Security users require relevant resource permissions.

About this task

Advanced Message Security users, that is users that are putting or getting Advanced Message Security protected messages, require:

- An OMVS segment associated with their user id
- Permissions for IRR.DIGTCERT.LISTRING or RDATA LIB
- Permissions for ICSF class CSFSERV and CSFKEYS profiles
- Permission to put to the SYSTEM.PROTECTION.ERROR.QUEUE

The Advanced Message Security task temporarily assumes the identity of its clients; that is, the task acts as a surrogate of the z/OS user ID of users of Advanced Message Security during the processing of IBM MQ messages to queues that are protected by Advanced Message Security.

In order for the task to assume the z/OS identity of a user, the client z/OS user ID must have a defined OMVS segment associated with its user profile.

As an administration aid, RACF provides the ability to define a default OMVS segment that may be associated with RACF user and group profiles. This default is used if the z/OS user ID or group profile does not have an OMVS segment explicitly defined. If you plan to have a large number of users using Advanced Message Security, you might choose to use this default rather than explicitly defining the OMVS segment for each user.

The *z/OS: Security Server RACF Security Administrator's Guide* contains the detailed procedure for defining default OMVS segments. Review the procedure as outlined in this publication to determine if the definition of default OMVS segments in RACF User and Group profiles is appropriate to your installation.

Procedure

1. Grant READ permission to the IRR.DIGTCERT.LISTRING profile in the FACILITY class:

- To grant READ permission to the IRR.DIGTCERT.LISTRING profile in the FACILITY class to all users, issue this command:

```
RDEFINE FACILITY IRR.DIGTCERT.LISTRING UACC(READ)
```

- To grant READ permission to the IRR.DIGTCERT.LISTRING profile in the FACILITY class on a per user basis, issue this command:

```
PERMIT IRR.DIGTCERT.LISTRING CLASS(FACILITY) ID(userid) ACCESS(READ)
```

where userid is the name of the Advanced Message Security user.

- Alternatively, use the RDATA LIB class to grant access to specific key rings. The RDATA LIB permissions take precedence over IRR.DIGTCERT.LISTRING permissions. For example:

```
PERMIT user.DRQ.AMS.KEYRING.LST CLASS(RDATA LIB) ID(user) ACC(READ)
```

2. If you are using ICSF-managed certificates and private keys, Advanced Message Security users require access to certain class CSFSERV and CSFKEYS profiles. This access is detailed in the following table:

Class	Profile	Permission
CSFSERV	CSFDSG	READ
CSFSERV	CSFPKE	READ
CSFSERV	CSFPKD	READ
CSFSERV	CSFDSV	READ

Table 64. Required user access to class CSFSERV and CSFKEYS profiles (continued)		
Class	Profile	Permission
CSFKEYS	ICSF PKDS Label	READ

- Applications that perform operations on queues with AMS policies defined need access to put messages to SYSTEM.PROTECTION.ERROR.QUEUE. Grant put access to the queue with these commands:

```
RDEFINE MQQUEUE hlq.SYSTEM.PROTECTION.ERROR.QUEUE UACC(NONE)
PERMIT hlq.SYSTEM.PROTECTION.ERROR.QUEUE CLASS(MQQUEUE) ID(userId) ACCESS(UPDATE)
```

where *hlq* can be either the queue manager name queue sharing group name, and *userId* is the application user ID.

What to do next

[“Create key rings for Advanced Message Security” on page 996](#)

Create key rings for Advanced Message Security

Certificates used by Advanced Message Security (AMS) for signing and encryption are stored in z/OS SAF key rings. You need to create these key rings and certificates before you can use AMS.

About this task

Advanced Message Security accesses certificates in the following key rings:

- A single key ring owned by the AMS address space user.
- Key rings owned by the individual users that send or receive messages on queues with AMS policies defined.

These key rings must all be named `drq.ams.keyring`.

There is more information on key rings and certificates used by AMS, and an example scenario, in [Using certificates on z/OS](#).

Follow these steps to create the key rings required by AMS, and connect certificates to the key rings. You must create the key ring owned by the AMS address space user before starting AMS. You can create the keys rings owned by the users that send or receive messages at any time.

Procedure

- Issue the following command to create a key ring owned by the AMS address space user:

```
RACDCERT ID(amsUser) ADDRING(drq.ams.keyring)
```

where *amsUser* is the user ID of the AMS address space.

- Create a key ring for each user that sends or receives messages protected by AMS by issuing the command in step 1 for each user ID.
- Connect the certificate authority (CA) certificate for the issuer of the user certificates to the key ring owned by the AMS address space user ID. Issue the following command:

```
RACDCERT ID(amsUser) CONNECT(CERTAUTH LABEL('caLabel') RING(drq.ams.keyring))
```

where *amsUser* is the user ID of the AMS address space, and *caLabel* is the label of the CA certificate.

If you are using RACF as your CA, and need to create a certificate authority certificate, follow the example in [Defining a local Certificate Authority certificate](#).

- If you are using privacy or confidentiality security policies to encrypt messages on queues protected by AMS, connect the certificates of message recipients to the key ring owned by the AMS address space user ID. Issue the following command:

```
RACDCERT ID(amsUser) CONNECT(ID(userId) LABEL('certLabel'))  
RING(dmq.ams.keyring) USAGE(SITE))
```

where *amsUser* is the user ID of the AMS address space, *userId* is the message recipient, and *certLabel* is the label of the user's certificate.

The USAGE (SITE) attribute prevents the private key from being accessible in the key ring.

If you are creating your own certificates with RACF, follow the example in [Creating a digital certificate with a private key](#) to create the certificate.

- Connect the certificates of each user that sends or receives messages protected by AMS to a key ring owned by the user. The certificate must be connected as the default certificate in the key ring. Issue the following command:

```
RACDCERT ID(userId) CONNECT(ID(userId) LABEL('certLabel'))  
RING(dmq.ams.keyring) DEFAULT USAGE(PERSONAL))
```

where *userId* is the user that is sending or receiving messages, and *certLabel* is the label of the user's certificate.

Notes:

- Steps “2” on [page 996](#) and “5” on [page 997](#) are not required if the application opens a queue only for output, and sends messages to queues protected by an AMS confidentiality policy.
- Steps “2” on [page 996](#) and “5” on [page 997](#) are not required if the application opens a queue only for input/browse, and receives messages from queues protected by an AMS integrity policy.

What to do next

[“Enable Advanced Message Security” on page 997](#)

Enable Advanced Message Security

Security policy capability for a queue manager is controlled by the SPLCAP parameter in the system parameter module.

About this task

Follow these steps to enable Advanced Message Security (AMS) for a single queue manager.

This task requires you to make a change to the system parameter module. See [“Tailor your system parameter module” on page 957](#) for more information on creating and customizing the system parameter module.

Procedure

- Set **SPLCAP** to YES in CSQ6SYSP. See [“Utilisation de CSQ6SYSP” on page 959](#) for more information on the CSQ6SYSP macro.
- Set the **AMSPROD** to either AMS, ADVANCED, or ADVANCEDVUE depending on your licence entitlement. See [using CSQ6USGP](#) for more information on the CSQ6USGP macro.
- Recompile the system parameter module.
- Restart the queue manager with the updated system parameter module. The AMS address space is started automatically when the queue manager starts.

Configuring the mqweb server

Use these topics as a step by step guide for configuring the mqweb server.

Related tasks

“Configuration de IBM MQ Console et de REST API” on page 883

Le serveur mqweb qui héberge IBM MQ Console et REST API est fourni avec une configuration par défaut. Pour utiliser l'un ou l'autre de ces composants, un certain nombre de tâches de configuration doivent être effectuées, telles que la configuration de la sécurité pour permettre aux utilisateurs de se connecter. Cette rubrique décrit toutes les options de configuration disponibles.

Création du serveur mqweb

Si vous avez installé le IBM MQ for z/OS UNIX System Services Web Componentset que vous souhaitez utiliser le IBM MQ Consoleou le REST API, vous devez créer et personnaliser le serveur mqweb.

Avant de commencer

Avant d'exécuter le script **crtmqweb** pour créer le serveur mqweb, définissez la variable d'environnement JAVA_HOME pour référencer une version 64 bits de Java sur votre système.

IBM MQ Console et administrative REST API requièrent SYSTEM.REST.REPLY.QUEUE de file d'attente à créer. Créez cette file d'attente à l'aide de l'exemple **CSQ4INSG** dans «Customize the initialization input data sets», à la page 950.



Avertissement : Lors du démarrage du serveur mqweb, si vous rencontrez le message d'erreur CWWKG0014E, comme indiqué dans la sortie suivante:

```
Launching_mqweb (MQM MVS/ESA V9 R2.0/wlp...) (en_US)
YAUDIT      CWWKE0001I: The server mqweb has been
launched.
           YWARNING  " CWWKF0009W: The server has not been configured to install any
features.
           YAUDIT    " CWWKF0011I: The mqweb server is ready to run a smarter planet.
The mqweb server started in 6.348 seconds.
           YERROR    " CWWKG0014E: The configuration parser detected an XML syntax
error while parsing the root of the configuration and the referenced configuration
documents.
                                     Error: An invalid XML character (Unicode: 0x4c) was found
in the prolog of the document.
                                     File: file:<your filepath>/servers/mqweb/server.xml Line:
1 Column: 1
```

vous devez vérifier le paramètre z/OS de AUTOCVT (convertir automatiquement les fichiers d'un jeu de codes à un autre) et ajuster la valeur comme requis en effectuant l'une des opérations suivantes.

Dans un terminal USS:

Exécutez la commande: `echo $_BPXK_AUTOCVT` pour afficher la valeur de cette variable d'environnement. Si la variable d'environnement n'est pas définie, aucune valeur n'est affichée.

Pour définir la variable d'environnement, voir [_BPXK environment variables](#).

A l'échelle du système:

L'exemple 6 de la page [Affichage de l'état de z/OS UNIX System Services \(OMVS\)](#) montre comment afficher la valeur de l'instruction AUTOCVT à l'échelle du système dans BPXPRMxx.

Pour définir la variable d'environnement à l'échelle du système, utilisez l'instruction [AUTOCVT](#) dans BPXPRMxx.

Si la variable d'environnement `_BPXX_AUTOCVT` est définie dans un terminal USS, elle remplace le paramètre système de l'instruction AUTOCVT dans BPXPRMxx.

Pourquoi et quand exécuter cette tâche

- Effectuez cette tâche une fois pour chaque système z/OS sur lequel vous souhaitez exécuter IBM MQ Console ou REST API.
- Pour utiliser le administrative REST API, vous avez besoin d'un serveur mqweb pour chaque version de IBM MQ en cours d'exécution. Par exemple, si vous exécutez IBM MQ 9.4.0, 9.3.5 et 9.3.0, vous avez besoin de trois serveurs mqweb différents.

- Vous devrez peut-être actualiser ou modifier la configuration du serveur lorsque vous effectuez une migration à partir d'une version précédente.

IBM MQ Console et REST API nécessitent la création d'un serveur WebSphere Liberty unique, appelé mqweb.

La configuration du serveur et les fichiers journaux sont tous stockés dans le répertoire utilisateur Liberty .

Le serveur mqweb doit être configuré avec un ID de produit (PID) sous lequel il s'exécute. Le PID est défini lors de la création du serveur mqweb. Utilisez le même PID que celui utilisé pour exécuter les gestionnaires de files d'attente locaux auxquels le serveur mqweb se connecte.

Remarque : Si les gestionnaires de files d'attente locaux s'exécutent sous plusieurs PID différents, choisissez-en un pour exécuter le serveur mqweb.

Pour plus d'informations sur les PID et leur utilisation sur z/OS, voir [Enregistrement de l'utilisation du produit avec les produits IBM MQ for z/OS](#).

Il est possible de modifier le PID sous lequel s'exécute le serveur mqweb, après sa création, à l'aide de la commande `setmqweb pid` .

Procédez comme suit pour créer le serveur mqweb:

Procédure

1. Déterminez le PID sous lequel s'exécute le serveur mqweb.
2. Choisissez un emplacement approprié pour l'annuaire d'utilisateurs Liberty .

L'ID utilisateur sous lequel s'exécute le serveur mqweb requiert un accès en lecture et en écriture à cet annuaire d'utilisateurs et à son contenu. Comme ce répertoire utilisateur contient des fichiers journaux, en plus de la configuration du serveur, créez ce répertoire dans un système de fichiers distinct.

Remarque : Il y a une quantité importante d'E-S de disque lorsque le serveur mqweb démarre. Pour réduire le temps nécessaire au démarrage du serveur mqweb, vérifiez que le système de fichiers z/OS UNIX de l'installation IBM MQ et le système de fichiers du répertoire utilisateur Liberty sont tous deux sysplex-aware ou montés en local sur le système sur lequel le serveur mqweb est en cours d'exécution.

3. Dans z/OS UNIX System Services, remplacez votre répertoire de travail en cours par `PathPrefix/web/bin` en exécutant la commande suivante:

```
cd PathPrefix/web/bin
```

où `PathPrefix` est le chemin d'installation de IBM MQ for z/OS UNIX System Services Components .

4. Créez le répertoire utilisateur Liberty qui contient le modèle de définition de serveur mqweb en exécutant le script `crtmqweb` .

Le format de la commande `crtmqweb` est le suivant:

```
crtmqweb user_directory -p pid_value
```

où :

répertoire_utilisateur

Correspond à l'annuaire d'utilisateurs Liberty choisi à l'étape «2», à la page 999. Ce paramètre est facultatif. Si ce paramètre n'est pas spécifié, un répertoire utilisateur Liberty par défaut de `/var/mqm/web/installation1` est utilisé.

valeur_Idéfinie

Indique le PID sous lequel s'exécute le serveur mqweb. Ce PID est celui que vous avez choisi à l'étape «1», à la page 999. `pid_value` est l'une des valeurs suivantes:

MQ

Le serveur mqweb s'exécute sous le PID IBM MQ for z/OS (5655-MQ9).

VUE

Le serveur mqweb s'exécute sous le PID IBM MQ for z/OS Value Unit Edition (VUE) (5655-VU9).

ADVANCEDVUE

Le serveur mqweb s'exécute sous le PID IBM MQ Advanced for z/OS VUE (5655-AV1),

Par exemple, pour créer le serveur mqweb avec le répertoire utilisateur Liberty /usr/mqweb et le PID IBM MQ Advanced for z/OS VUE (5655-AV1), exécutez la commande suivante:

```
./crtmqweb /usr/mqweb -p ADVANCEDVUE
```

5. Modifiez la propriété des répertoires et des fichiers dans le répertoire utilisateur Liberty afin qu'ils appartiennent à l'ID utilisateur et au groupe sous lesquels s'exécute le serveur mqweb, à l'aide de la commande suivante:

```
chown -R userid:group path
```

Pour accorder au groupe un accès en écriture au chemin, exécutez la commande suivante:

```
chmod -R 770 path
```

Que faire ensuite

«Creating a procedure for the mqweb server», à la page 1000

Tâches associées

«Configuration de IBM MQ Console et de REST API», à la page 883

Le serveur mqweb qui héberge IBM MQ Console et REST API est fourni avec une configuration par défaut. Pour utiliser l'un ou l'autre de ces composants, un certain nombre de tâches de configuration doivent être effectuées, telles que la configuration de la sécurité pour permettre aux utilisateurs de se connecter. Cette rubrique décrit toutes les options de configuration disponibles.

Creating a procedure for the mqweb server

If you installed the IBM MQ for z/OS UNIX System Services Web Components, and want to use the IBM MQ Console, or the REST API, you need to create a cataloged procedure to start the mqweb server. The mqweb server is a Liberty server that hosts the IBM MQ Console and the REST API.

- You need to perform this task once for each z/OS system where you want to run the IBM MQ Console or REST API.
- You need a mqweb server for each version of IBM MQ that is running. For example, a started task called MQWB0910 for queue managers at IBM MQ for z/OS 9.1.0 and a started task called MQWB0905 for queue managers at IBM MQ for z/OS 9.0.5.

If you have only one queue manager on the z/OS system, you can run a single Liberty server started task, and change the libraries it uses when you migrate your queue manager.

- You might need to modify the cataloged procedure when migrating from a previous version.

Carry out the following procedure to create a cataloged procedure:

1. Copy the sample started task procedure `th1qua1.SCSQPROC (CSQ4WEBS)` to your procedure library.

Name the procedure according to the standards of your enterprise.

For example `MQWB0910`, indicating that this is the cataloged procedure for the IBM MQ for z/OS 9.1.0 mqweb server.

2. Tailor the procedure to your requirements using the instructions in the sample procedure `CSQ4WEBS`.

Note that the Liberty user directory is the directory specified when the `crtmqweb` script was run to create the mqweb server definition.

See “[Création du serveur mqweb](#)” on page 998 for details.

Note: Ensure that you specify **Caps off** when you edit the member, as the file has lowercase data.

3. Authorize the procedure to run under your external security manager.
4. Use IBM Workload Manager (WLM) to classify this address space.

The mqweb server is an IBM MQ application, and users interact with this application. The application does not need to be high importance in WLM, and a service class of **STCUSER** might be suitable.

What to do next

Follow the steps in [“Configuration de base pour le serveur mqweb” on page 883](#) to finish configuring the mqweb server.

Related tasks

[“Configuration de IBM MQ Console et de REST API” on page 883](#)

Le serveur mqweb qui héberge IBM MQ Console et REST API est fourni avec une configuration par défaut. Pour utiliser l'un ou l'autre de ces composants, un certain nombre de tâches de configuration doivent être effectuées, telles que la configuration de la sécurité pour permettre aux utilisateurs de se connecter. Cette rubrique décrit toutes les options de configuration disponibles.

▶ z/OS

Testing a queue manager on z/OS

When you have customized or migrated your queue manager, you can test it by running the installation verification programs and some of the sample applications shipped with IBM MQ for z/OS.

About this task

After you have installed and customized IBM MQ for z/OS, you can use the supplied installation verification program, CSQ4IVP1, to confirm that IBM MQ for z/OS is operational.

The basic installation verification program CSQ4IVP1 tests non-shared queues and verifies the base IBM MQ without using the C, COBOL, or CICS samples.

After running the basic installation verification, you can test for shared queues by using CSQ4IVP1 with different queues, and also test that Db2 and the coupling facility are set up correctly. To confirm that distributed queuing is operational, you can use the supplied installation verification program, CSQ4IVPX,

CSQ4IVP1 is supplied as a load module, and provides a set of procedural sample applications as source modules that demonstrate typical uses of the Message Queue Interface (MQI). You can use these source modules to test different programming language environments. You can compile and link-edit whichever of the other samples are appropriate to your installation by using the supplied sample JCL supplied.

Procedure

- For information on how to test your queue manager on z/OS, see the following subtopics:
 - [“Running the basic installation verification program” on page 1002](#)
 - [“Testing for queue sharing groups” on page 1005](#)
 - [“Testing for distributed queuing” on page 1006](#)
 - [“Testing for C, C++, COBOL, PL/I, and CICS programs with IBM MQ for z/OS” on page 1009](#)

Related concepts

[IBM MQ for z/OS concepts](#)

Related tasks

[Planning your IBM MQ environment on z/OS](#)

[“Configuring queue managers on z/OS” on page 930](#)

Use these instructions to configure queue managers on IBM MQ for z/OS.

[Administering IBM MQ for z/OS](#)

Running the basic installation verification program

After you have installed and customized IBM MQ, you can use the supplied installation verification program, CSQ4IVP1, to confirm that IBM MQ is operational.

The basic installation verification program is a batch assembler IVP that verifies the base IBM MQ without using the C, COBOL, or CICS samples.

The Batch Assembler IVP is link-edited by SMP/E and the load modules are shipped in library thlqual.SCSQLOAD.

After you have completed both the SMP/E APPLY step and the customization steps, run the Batch Assembler IVP.

See these sections for further details:

- [Overview of the CSQ4IVP1 application](#)
- [Preparing to run CSQ4IVP1](#)
- [Running CSQ4IVP1](#)
- [Checking the results of CSQ4IVP1](#)

Overview of the CSQ4IVP1 application

CSQ4IVP1 is a batch application that connects to your IBM MQ subsystem and performs these basic functions:

- Issues IBM MQ calls
- Communicates with the command server
- Verifies that triggering is active
- Generates and deletes a dynamic queue
- Verifies message expiry processing
- Verifies message commit processing

Preparing to run CSQ4IVP1

Before you run CSQ4IVP1:

1. Check that the IVP entries are in the CSQINP2 data set concatenation in the queue manager startup program. The IVP entries are supplied in member thlqual.SCSQPROC(CSQ4IVPQ). If not, add the definitions supplied in thlqual.SCSQPROC(CSQ4IVPQ) to your CSQINP2 concatenation. If the queue manager is currently running, you need to restart it so that these definitions can take effect.
2. The sample JCL, CSQ4IVPR, required to run the installation verification program is in library thlqual.SCSQPROC.

Customize the CSQ4IVPR JCL with the high-level qualifier for the IBM MQ libraries, the national language you want to use, the four-character IBM MQ queue manager name, and the destination for the job output.

3. Update RACF to allow CSQ4IVP1 to access its resources if IBM MQ security is active.

To run CSQ4IVP1 when IBM MQ security is enabled, you need a RACF user ID with authority to access the objects. For details of defining resources to RACF, see [Setting up security on z/OS](#). The user ID that runs the IVP must have the following access authority:

Authority	Profile	Class
READ	ssid.DISPLAY.PROCESS	MQCMDS
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE

Authority	Profile	Class
UPDATE	ssid.SYSTEM.COMMAND.REPLY.MODEL	MQQUEUE
UPDATE	ssid.CSQ4IVP1.**	MQQUEUE
READ	ssid.BATCH	MQCONN

These requirements assume that all IBM MQ security is active. The RACF commands to activate IBM MQ security are shown in [Figure 98 on page 1003](#). This example assumes that the queue manager name is CSQ1 and that the user ID of the person running sample CSQ4IVP1 is TS101.

```
RDEFINE MQCMDS CSQ1.DISPLAY.PROCESS
PERMIT CSQ1.DISPLAY.PROCESS CLASS(MQCMDS) ID(TS101) ACCESS(READ)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.INPUT
PERMIT CSQ1.SYSTEM.COMMAND.INPUT CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.REPLY.MODEL
PERMIT CSQ1.SYSTEM.COMMAND.REPLY.MODEL CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.CSQ4IVP1.**
PERMIT CSQ1.CSQ4IVP1.** CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQCONN CSQ1.BATCH
PERMIT CSQ1.BATCH CLASS(MQCONN) ID(TS101) ACCESS(READ)
```

Figure 98. RACF commands for CSQ4IVP1

Running CSQ4IVP1

When you have completed these steps, start your queue manager. If the queue manager is already running and you have changed CSQINP2, you must stop the queue manager and restart it.


The IVP runs as a batch job. Customize the job card to meet the submission requirements of your installation.

Checking the results of CSQ4IVP1

The IVP is split into 10 stages; each stage must complete with a zero completion code before the next stage is run. The IVP generates a report, listing:

- The name of queue manager that is being connected to.
- A one-line message showing the completion code and the reason code returned from each stage.
- A one-line informational message where appropriate.

A sample report is provided in [Figure 99 on page 1005](#)

 For an explanation of the completion and reason codes, see the [IBM MQ for z/OS messages, completion, and reason codes](#).

Some stages have more than one IBM MQ call and, in the event of failure, a message is issued indicating the specific IBM MQ call that returned the failure. Also, for some stages the IVP puts explanatory and diagnostic information into a comment field.

The IVP job requests exclusive control of certain queue manager objects and therefore should be single threaded through the system. However, there is no limit to the number of times the IVP can be run against your queue manager.

The functions performed by each stage are:

Stage 1

Connect to the queue manager by issuing the MQCONN API call.

Stage 2

Determine the name of the system-command input queue used by the command server to retrieve request messages. This queue receives display requests from Stage 5.

To do this, the sequence of calls is:

1. Issue an MQOPEN call, specifying the queue manager name, to open the queue manager object.
2. Issue an MQINQ call to find out the name of the system-command input queue.
3. Issue an MQINQ call to find out about various queue manager event switches.
4. Issue an MQCLOSE call to close the queue manager object.

On successful completion of this stage, the name of the system-command input queue is displayed in the comment field.

Stage 3

Open an initiation queue using an **MQOPEN** call.

This queue is opened at this stage in anticipation of a trigger message, which arrives as a result of the command server replying to the request from Stage 5. The queue must be opened for input to meet the triggering criteria.

Stage 4

Create a permanent dynamic queue using the CSQ4IVP1.MODEL queue as a model. The dynamic queue has the same attributes as the model from which it was created. This means that when the replies from the command server request in Stage 5 are written to this queue, a trigger message is written to the initiation queue opened in Stage 3.

Upon successful completion of this stage, the name of the permanent dynamic queue is indicated in the comment field.

Stage 5

Issue an MQPUT1 request to the command server command queue.

A message of type MQMT_REQUEST is written to the system-command input queue requesting a display of process CSQ4IVP1. The message descriptor for the message specifies the permanent dynamic queue created in Stage 4 as the reply-to queue for the command server's response.

Stage 6

Issue an **MQGET** request from the initiation queue. At this stage, a GET WAIT with an interval of 1 minute is issued against the initiation queue opened in Stage 3. The message returned is expected to be the trigger message generated by the command server's response messages being written to the reply-to queue.

Stage 7

Delete the permanent dynamic queue created in Stage 4. As the queue still has messages on it, the MQCO_PURGE_DELETE option is used.

Stage 8

1. Open a dynamic queue.
2. MQPUT a message with an expiry interval set.
3. Wait for the message to expire.
4. Attempt to MQGET the expired message.
5. MQCLOSE the queue.

Stage 9

1. Open a dynamic queue.
2. MQPUT a message.
3. Issue MQCMIT to commit the current unit of work.

4. MQGET the message.
5. Issue MQBACK to backout the message.
6. MQGET the same message and ensure that the backout count is set to 1.
7. Issue MQCLOSE to close the queue.

Stage 10

Disconnect from the queue manager using **MQDISC**.

After running the IVP, you can delete any objects that you no longer require.

If the IVP does not run successfully, try each step manually to find out which function is failing.

```

DATE : 2005.035                IBM MQ for z/OS - V6                PAGE : 0001
INSTALLATION VERIFICATION PROGRAM
PARAMETERS ACCEPTED. PROGRAM WILL CONNECT TO : CSQ1
,OBJECT QUALIFER : CSQ4IVP1
INSTALLATION VERIFICATION BEGINS :
STAGE 01 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR BRIDGE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS EXCP FOR CHANNEL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR SSL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR INHIBITED EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR LOCAL EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR PERFORMANCE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR REMOTE EVENTS
STAGE 02 INFO: QMGR EVENT SWITCH IS OFF FOR START/STOP EVENTS
STAGE 02 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 SYSTEM.COMMAND.INPUT
STAGE 03 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 04 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9810EFEAC8980
STAGE 05 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 06 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 07 COMPLETE. COMPCODE : 0000 REASON CODE : 0000
STAGE 08 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9810F0070E645
STAGE 09 COMPLETE. COMPCODE : 0000 REASON CODE : 0000 CSQ4IVP1.BAB9812BA8706803
STAGE 10 COMPLETE. COMPCODE : 0000 REASON CODE : 0000>>>>>>>>>> END OF REPORT <<<<<<<<<<<<

```

Figure 99. Sample report from CSQ4IVP1

Testing for queue sharing groups

The basic installation verification program CSQ4IVP1 tests non-shared queues.

CSQ4IVP1 can be used whether the queue manager is a member of a queue sharing group or not. After running the basic IVP, you can test for shared queues by using the CSQ4IVP1 installation verification program with different queues. Also this tests that Db2 and the coupling facility are set up correctly.

Preparing to run CSQ4IVP1 for a queue sharing group

Before you run CSQ4IVP1:

1. Add the coupling facility structure that the IVP uses to your CFRM policy data set, as described in “[Set up the coupling facility](#)” on [page 988](#). The supplied samples use a structure called APPLICATION1, but you can change this if you want.
2. Check that the IVP entries are in the CSQINP2 data set concatenation in the queue manager startup program. The IVP entries are supplied in member thlqual.SCSQPROC(CSQ4IVPG). If they are not, add the definitions supplied in thlqual.SCSQPROC(CSQ4IVPG) to your CSQINP2 concatenation. If the queue manager is currently running, you need to restart it so that these definitions can take effect.
3. Change the name of the coupling facility structure used in thlqual.SCSQPROC(CSQ4IVPG) if necessary.
4. The sample JCL, CSQ4IVPS, required to run the installation verification program for a queue sharing group is in library thlqual.SCSQPROC.

Customize the CSQ4IVPS JCL with the high-level qualifier for the IBM MQ libraries, the national language you want to use, the four-character IBM MQ queue manager name, and the destination for the job output.

5. Update RACF to allow CSQ4IVP1 to access its resources if IBM MQ security is active.

To run CSQ4IVP1 when IBM MQ security is enabled, you need a RACF user ID with authority to access the objects. For details of defining resources to RACF, see [Setting up security on z/OS](#). The user ID that runs the IVP must have the following access authority in addition to that required to run the basic IVP:

Authority	Profile	Class
UPDATE	ssid.CSQ4IVPG.**	MQQUEUE

These requirements assume that all IBM MQ security is active. The RACF commands to activate IBM MQ security are shown in [Figure 100](#) on page 1006. This example assumes that the queue manager name is CSQ1 and that the user ID of the person running sample CSQ4IVP1 is TS101.

```
RDEFINE MQQUEUE CSQ1.CSQ4IVPG.**
PERMIT CSQ1.CSQ4IVPG.** CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)
```

Figure 100. RACF commands for CSQ4IVP1 for a queue sharing group

Running CSQ4IVP1 for a queue sharing group

When you have completed these steps, start your queue manager. If the queue manager is already running and you have changed CSQINP2, you must stop the queue manager and restart it.

The IVP runs as a batch job. Customize the job card to meet the submission requirements of your installation.

Checking the results of CSQ4IVP1 for a queue sharing group

The IVP for queue sharing groups works in the same way as the basic IVP, except that the queues that are created are called CSQIVPG. xx. Follow the instructions given in [“Checking the results of CSQ4IVP1”](#) on page 1003 to check the results of the IVP for queue sharing groups.

Testing for distributed queuing

You can use the supplied installation verification program, CSQ4IVPX, to confirm that distributed queuing is operational.

Overview of CSQ4IVPX job

CSQ4IVPX is a batch job that starts the channel initiator and issues the IBM MQ DISPLAY CHINIT command. This verifies that all major aspects of distributed queuing are operational, while avoiding the need to set up channel and network definitions.

Preparing to run CSQ4IVPX

Before you run CSQ4IVPX:

1. The sample JCL, CSQ4IVPX, required to run the installation verification program is in library thlqual.SCSQPROC.

Customize the CSQ4IVPX JCL with the high-level qualifier for the IBM MQ libraries, the national language you want to use, the four-character queue manager name, and the destination for the job output.

2. Update RACF to allow CSQ4IVPX to access its resources if IBM MQ security is active. To run CSQ4IVPX when IBM MQ security is enabled, you need a RACF user ID with authority to access the objects. For

details of defining resources to RACF, see [Setting up security on z/OS](#) . The user ID that runs the IVP must have the following access authority:

Authority	Profile	Class
CONTROL	ssid.START.CHINIT and ssid.STOP.CHINIT	MQCMDS
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.CSQUTIL.*	MQQUEUE
READ	ssid.BATCH	MQCONN
READ	ssid.DISPLAY.CHINIT	MQCMDS

These requirements assume that the connection security profile ssid.CHIN has been defined (as shown in [Connection security profiles for the channel initiator](#)), and that all IBM MQ security is active. The RACF commands to do this are shown in [Figure 101 on page 1008](#). This example assumes that:

- The queue manager name is CSQ1
- The user ID of the person running sample CSQ4IVPX is TS101
- The channel initiator address space is running under the user ID CSQ1MSTR

3. Update RACF to allow the channel initiator address space the following access authority:

Authority	Profile	Class
READ	ssid.CHIN	MQCONN
UPDATE	ssid.SYSTEM.COMMAND.INPUT	MQQUEUE
UPDATE	ssid.SYSTEM.CHANNEL.INITQ	MQQUEUE
UPDATE	ssid.SYSTEM.CHANNEL.SYNCQ	MQQUEUE
ALTER	ssid.SYSTEM.CLUSTER.COMMAND.QUEUE	MQQUEUE
UPDATE	ssid.SYSTEM.CLUSTER.TRANSMIT.QUEUE	MQQUEUE
ALTER	ssid.SYSTEM.CLUSTER.REPOSITORY.QUEUE	MQQUEUE
CONTROL	ssid.CONTEXT.**	MQADMIN

The RACF commands to do this are also shown in [Figure 101 on page 1008](#).

```

RDEFINE MQCMDS CSQ1.DISPLAY.DQM
PERMIT CSQ1.DISPLAY.DQM CLASS(MQCMDS) ID(TS101) ACCESS(READ)

RDEFINE MQCMDS CSQ1.START.CHINIT
PERMIT CSQ1.START.CHINIT CLASS(MQCMDS) ID(TS101) ACCESS(CONTROL)

RDEFINE MQCMDS CSQ1.STOP.CHINIT
PERMIT CSQ1.STOP.CHINIT CLASS(MQCMDS) ID(TS101) ACCESS(CONTROL)

RDEFINE MQQUEUE CSQ1.SYSTEM.COMMAND.INPUT
PERMIT CSQ1.SYSTEM.COMMAND.INPUT CLASS(MQQUEUE) ID(TS101,CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CSQUTIL.*
PERMIT CSQ1.SYSTEM.CSQUTIL.* CLASS(MQQUEUE) ID(TS101) ACCESS(UPDATE)

RDEFINE MQCONN CSQ1.BATCH
PERMIT CSQ1.BATCH CLASS(MQCONN) ID(TS101) ACCESS(READ)

RDEFINE MQCONN CSQ1.CHIN
PERMIT CSQ1.CHIN CLASS(MQCONN) ID(CSQ1MSTR) ACCESS(READ)

RDEFINE MQQUEUE CSQ1.SYSTEM.CHANNEL.SYNCQ
PERMIT CSQ1.SYSTEM.CHANNEL.SYNCQ CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.COMMAND.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.COMMAND.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(ALTER)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.TRANSMIT.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.TRANSMIT.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQQUEUE CSQ1.SYSTEM.CLUSTER.REPOSITORY.QUEUE
PERMIT CSQ1.SYSTEM.CLUSTER.REPOSITORY.QUEUE CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(ALTER)

RDEFINE MQQUEUE CSQ1.SYSTEM.CHANNEL.INITQ
PERMIT CSQ1.SYSTEM.CHANNEL.INITQ CLASS(MQQUEUE) ID(CSQ1MSTR) ACCESS(UPDATE)

RDEFINE MQADMIN CSQ1.CONTEXT.**
PERMIT CSQ1.CONTEXT.** CLASS(MQADMIN) ID(CSQ1MSTR) ACCESS(CONTROL)

```

Figure 101. RACF commands for CSQ4IVPX

Running CSQ4IVPX

When you have completed these steps, start your queue manager.

The IVP runs as a batch job. Customize the job card to meet the submission requirements of your installation.

Checking the results of CSQ4IVPX

CSQ4IVPX runs the CSQUTIL IBM MQ utility to issue three MQSC commands. The SYSPRINT output data set should look like [Figure 102 on page 1009](#), although details might differ depending on your queue manager attributes.

- You should see the commands **(1)** each followed by several messages.
- The last message from each command should be "CSQ9022I ... NORMAL COMPLETION" **(2)**.
- The job as a whole should complete with return code zero **(3)**.


```

CSQU000I CSQUTIL IBM MQ for z/OS - V6
CSQU001I CSQUTIL Queue Manager Utility - 2005-05-09 09:06:48
COMMAND
CSQU127I CSQUTIL Executing COMMAND using input from CSQUCMD data set
CSQU120I CSQUTIL Connecting to queue manager CSQ1
CSQU121I CSQUTIL Connected to queue manager CSQ1
CSQU055I CSQUTIL Target queue manager is CSQ1
START CHINIT
(1)
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQM138I +CSQ1 CSQMSCHI CHANNEL INITIATOR STARTING
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' START CHINIT' NORMAL COMPLETION
(2)
DISPLAY CHINIT
(1)
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMDDQM DISPLAY CHINIT COMMAND ACCEPTED
CSQN205I COUNT= 12, RETURN=00000000, REASON=00000000
CSQX830I +CSQ1 CSQXRQDM Channel initiator active
CSQX002I +CSQ1 CSQXRQDM Queue sharing group is QSG1
CSQX831I +CSQ1 CSQXRQDM 8 adapter subtasks started, 8 requested
CSQX832I +CSQ1 CSQXRQDM 5 dispatchers started, 5 requested
CSQX833I +CSQ1 CSQXRQDM 0 SSL server subtasks started, 0 requested
CSQX840I +CSQ1 CSQXRQDM 0 channel connections current, maximum 200
CSQX841I +CSQ1 CSQXRQDM 0 channel connections active, maximum 200,
including 0 paused
CSQX842I +CSQ1 CSQXRQDM 0 channel connections starting,
0 stopped, 0 retrying
CSQX836I +CSQ1 Maximum channels - TCP/IP 200, LU 6.2 200
CSQX845I +CSQ1 CSQXRQDM TCP/IP system name is TCP/IP
CSQX848I +CSQ1 CSQXRQDM TCP/IP listener INDISP=QMGR not started
CSQX848I +CSQ1 CSQXRQDM TCP/IP listener INDISP=GROUP not started
CSQX849I +CSQ1 CSQXRQDM LU 6.2 listener INDISP=QMGR not started
CSQX849I +CSQ1 CSQXRQDM LU 6.2 listener INDISP=GROUP not started
CSQ9022I +CSQ1 CSQXCRPS ' DISPLAY CHINIT' NORMAL COMPLETION
(2)
STOP CHINIT
(1)
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000004
CSQM137I +CSQ1 CSQMTCHI STOP CHINIT COMMAND ACCEPTED
CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
CSQ9022I +CSQ1 CSQXCRPS ' STOP CHINIT' NORMAL COMPLETION
(2)
CSQU057I CSQUCMDS 3 commands read
CSQU058I CSQUCMDS 3 commands issued and responses received, 0 failed
CSQU143I CSQUTIL 1 COMMAND statements attempted
CSQU144I CSQUTIL 1 COMMAND statements executed successfully
CSQU148I CSQUTIL Utility completed, return code=0
(3)

```

Figure 102. Example output from CSQ4IVPX

z/OS Testing for C, C++, COBOL, PL/I, and CICS programs with IBM MQ for z/OS

You can test for C, C++, COBOL, PL/I, or CICS, using the sample applications supplied with IBM MQ.

The IVP (CSQ4IVP1) is supplied as a load module, and provides the samples as source modules. You can use these source modules to test different programming language environments.

For more information about sample applications, see [Sample programs for IBM MQ for z/OS](#).

z/OS Setting up communications with other queue managers on z/OS

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

About this task

To define your distributed-queuing requirements, you need to define the following items:

- The channel initiator procedures and data sets
- The channel definitions
- The queues and other objects
- Access security

If you are using queue sharing groups, see [Distributed queuing and queue sharing groups](#).

For additional points to consider when you are preparing to set up distributed queuing with IBM MQ for z/OS, see [“Considerations for using distributed queuing on z/OS” on page 1010](#).

Procedure

To enable distributed queuing, complete the following steps:

- Customize the distributed queuing facility and define the IBM MQ objects required as described in [Defining system objects](#) and [“Preparing to customize queue managers on z/OS” on page 931](#).
- Define access security as described in [Security considerations for the channel initiator on z/OS](#).
- Set up your communications as described in [“Setting up communication for z/OS” on page 1029](#).

Related concepts

[“Setting up IBM MQ for z/OS” on page 935](#)

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Related tasks

[“Configuration de la file d'attente répartie” on page 212](#)

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ , y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Considerations for using distributed queuing on z/OS

Points to consider when you are preparing to use distributed queuing on z/OS.

If you are using queue sharing groups, see [Distributed queuing and queue sharing groups](#).

Operator messages

Because the channel initiator uses a number of asynchronously operating dispatchers, operator messages might occur on the log out of chronological sequence.

Channel operation commands

Channel operation commands generally involve two stages. When the command syntax has been checked and the existence of the channel verified, a request is sent to the channel initiator. Message [CSQM134I](#) or [CSQM137I](#) is sent to the command issuer to indicate the completion of the first stage. When the channel initiator has processed the command, further messages indicating its success or otherwise are sent to the command issuer along with message [CSQ9022I](#) or [CSQ9023E](#). Any error messages generated could also be sent to the z/OS console.

All cluster commands except **DISPLAY CLUSQMgr**, however, work asynchronously. Commands that change object attributes update the object and send a request to the channel initiator. Commands for working with clusters are checked for syntax and a request is sent to the channel initiator. In both cases, message [CSQM130I](#) is sent to the command issuer indicating that a request has been sent. This message is followed by message [CSQ9022I](#) to indicate that the command has completed successfully, in that a request has been sent. It does not indicate that the cluster request has completed successfully. The requests sent to the channel initiator are processed asynchronously, along with cluster requests received

from other members of the cluster. In some cases, these requests must be sent to the whole cluster to determine if they are successful or not. Any errors are reported to the z/OS on the system where the channel initiator is running. They are not sent to the command issuer.

Undelivered-message queue

A Dead Letter handler is provided with IBM MQ for z/OS. For more information, see [The dead-letter queue handler utility \(CSQUDLQH\)](#).

Queues in use

MCAs for receiver channels can keep the destination queues open even when messages are not being transmitted. This behavior results in the queues appearing to be 'in use'.

Security changes

If you change security access for a user ID, the change might not take effect immediately. For more information, see [Security considerations for the channel initiator on z/OS](#), [Profiles for queue security](#), and [“Implement your ESM security controls”](#) on page 949.

Communications stopped - TCP

If TCP is stopped for some reason and then restarted, the IBM MQ for z/OS TCP listener waiting on a TCP port is stopped.

Automatic channel-reconnect allows the channel initiator to detect that TCP/IP is unavailable and to automatically restart the TCP/IP listener when TCP/IP returns. This automatic restart alleviates the need for operations staff to notice the problem with TCP/IP and manually restart the listener. While the listener is out of action, the channel initiator can also be used to try the listener again at the interval specified by LSTRTMR. These attempts can continue until TCP/IP returns and the listener successfully restarts automatically. For more information about LSTRTMR, see [ALTER QMGR](#) and [Distributed queuing messages \(CSQX...\)](#).

Communications stopped - LU6.2

If APPC is stopped, the listener is also stopped. Again, in this case, the listener automatically tries again at the LSTRTMR interval so that, if APPC restarts, the listener can restart too.

If the Db2 fails, shared channels that are already running continue to run, but any new channel start requests fail. When the Db2 is restored new requests are able to complete.

z/OS Automatic Restart Management (ARM)

Automatic restart management (ARM) is a z/OS recovery function that can improve the availability of specific batch jobs or started tasks (for example, subsystems). It can therefore result in a faster resumption of productive work.

To use ARM, you must set up your queue managers and channel initiators in a particular way to make them restart automatically. For more information, see [Using the z/OS Automatic Restart Manager \(ARM\)](#).

Related concepts

[“Setting up IBM MQ for z/OS”](#) on page 935

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Related tasks

[“Configuration de la file d'attente répartie”](#) on page 212

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ , y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

Defining IBM MQ objects on z/OS

On z/OS, use one of the IBM MQ command input methods to define IBM MQ objects.

For more information about defining objects, see [“Monitoring and controlling channels on z/OS” on page 1013](#).

Transmission queues and triggering channels

Define the following:

- A local queue with the usage of XMITQ for each sending message channel.
- Remote queue definitions.

A remote queue object has three distinct uses, depending upon the way the name and content are specified:


- Remote queue definition
- Queue manager alias definition
- Reply-to queue alias definition

These three ways are shown in [Three ways of using the remote queue definition object](#).

Use the TRIGDATA field on the transmission queue to trigger the specified channel. For example:

```
DEFINE QLOCAL(MYXMITQ) USAGE(XMITQ) TRIGGER +  
INITQ(SYSTEM.CHANNEL.INITQ) TRIGDATA(MYCHANNEL)  
DEFINE CHL(MYCHANNEL) CHLTYPE(SDR) TRPTYPE(TCP) +  
XMITQ(MYXMITQ) CONNAME('9.20.9.30(1555)')
```

The supplied sample CSQ4INXD gives additional examples of the necessary definitions.

 Loss of connectivity to the CF structure where the synchronization queue for shared channels is defined, or similar problems, might temporarily prevent a channel from starting. After problem resolution, if you are using a trigger type of FIRST and the channel fails to start when it is triggered, you must start the channel manually. If you want to automatically start triggered channels after problem resolution, consider setting the queue manager TRIGINT attribute to a value other than the default. Setting the TRIGINT attribute to a value other than the default causes the channel initiator to retry starting the channel periodically while there are messages on the transmission queue.

Synchronization queue

DQM requires a queue for use with sequence numbers and logical units of work identifiers (LUWID). You must ensure that a queue is available with the name SYSTEM.CHANNEL.SYNCQ (see [Planification sur z/OS](#)). This queue must be available otherwise the channel initiator cannot start.

Make sure that you define this queue using INDXTYPE(MSGID). This attribute improves the speed at which they can be accessed.

Channel command queues

You need to ensure that a channel command queue exists for your system with the name SYSTEM.CHANNEL.INITQ.

If the channel initiator detects a problem with the SYSTEM.CHANNEL.INITQ, it is unable to continue normally until the problem is corrected. The problem could be one of the following:

- The queue is full
- The queue is not enabled for put
- The page set that the queue is on is full
- The channel initiator does not have the correct security authorization to the queue

If the definition of the queue is changed to GET(DISABLED) while the channel initiator is running, the initiator is unable to get messages from the queue, and terminates.

Starting the channel initiator

Triggering is implemented using the channel initiator. On IBM MQ for z/OS, the initiator is started with the MQSC command `START CHINIT`.

Stopping the channel initiator

The channel initiator is stopped automatically when you stop the queue manager. If you need to stop the channel initiator but not the queue manager, you can use the MQSC command `STOP CHINIT`.

Monitoring and controlling channels on z/OS

Use the DQM commands and panels to create, monitor, and control the channels to remote queue managers.

Each z/OS queue manager has a DQM program (the *channel initiator*) for controlling interconnections to remote queue managers using native z/OS facilities.

The implementation of these panels and commands on z/OS is integrated into the operations and control panels and the MQSC commands. No differentiation is made in the organization of these two sets of panels and commands.

You can also enter commands using Programmable Command Format (PCF) commands. See [Automating administration tasks](#) for information about using these commands.

The information in this section applies in all cases where the channel initiator is used for distributed queuing. It applies whether you are using queue sharing groups, or intra-group queuing.

The DQM channel control function

For an overview of the distributed queue management model, see [“Envoi et réception de messages” on page 237](#).

The channel control function consists of panels, commands and programs, two synchronization queues, channel command queues, and the channel definitions. This topic is a brief description of the components of the channel control function.

- The channel definitions are held as objects in page set zero or in Db2, like other IBM MQ objects in z/OS.
- You use the operations and control panels, MQSC commands, or PCF commands to:
 - Create, copy, display, alter, and delete channel definitions
 - Start and stop channel initiators and listeners
 - Start, stop, and ping channels, reset channel sequence numbers, and resolve in-doubt messages when links cannot be re-established
 - Display status information about channels
 - Display information about DQM

In particular, you can use the CSQINPX initialization input data set to issue your MQSC commands. This set can be processed every time you start the channel initiator. For more information, see [Initialization commands](#).

- There are two queues (SYSTEM.CHANNEL.SYNCQ and SYSTEM.QSG.CHANNEL.SYNCQ) used for channel re-synchronization purposes. Define these queues with `INDXTYPE(MSGID)` for performance reasons.
- The channel command queue (SYSTEM.CHANNEL.INITQ) is used to hold commands for channel initiators, channels, and listeners.
- The channel control function program runs in its own address space, separate from the queue manager, and comprises the channel initiator, listeners, MCAs, trigger monitor, and command handler.

- For queue sharing groups and shared channels, see [Shared queues and queue sharing groups](#).
- For intra-group queuing, see [Intra-group queuing](#)

Managing your channels on z/OS

Use the links in the following table for information about how to manage your channels, channel initiators, and listeners:

<i>Table 65. Channel tasks</i>	
Task to be performed	MQSC command
Define a channel	DEFINE CHANNEL
Alter a channel definition	ALTER CHANNEL
Display a channel definition	DISPLAY CHANNEL
Delete a channel definition	DELETE CHANNEL
Start a channel initiator	START CHINIT
Stop a channel initiator	STOP CHINIT
Display channel initiator information	DISPLAY CHINIT
Start a channel listener	START LISTENER
Stop a channel listener	STOP LISTENER
Start a channel	START CHANNEL
Test a channel	PING CHANNEL
Reset message sequence numbers for a channel	RESET CHANNEL
Resolve in-doubt messages on a channel	RESOLVE CHANNEL
Stop a channel	STOP CHANNEL
Display channel status	DISPLAY CHSTATUS
Display cluster channels	DISPLAY CLUSQMGR

Related concepts

[“Using the panels and the commands” on page 1015](#)

You can use the MQSC commands, the PCF commands, or the operations and control panels to manage DQM.

[“Setting up IBM MQ for z/OS” on page 935](#)

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

[“Setting up communication for z/OS” on page 1029](#)

When a distributed-queuing management channel is started, it tries to use the connection specified in the channel definition. To succeed, it is necessary for the connection to be defined and available. This section explains how to define a connection.

[“Preparing IBM MQ for z/OS for DQM with queue sharing groups” on page 1034](#)

Use the instructions in this section to configure distributed queuing with queue sharing groups on IBM MQ for z/OS.

[“Setting up communication for IBM MQ for z/OS using queue sharing groups” on page 1038](#)

When a distributed-queuing management channel is started, it attempts to use the connection specified in the channel definition. For this attempt to succeed, it is necessary for the connection to be defined and available.

Related tasks

[“Configuration de la file d'attente répartie” on page 212](#)

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ, y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

[“Setting up communications with other queue managers on z/OS” on page 1009](#)

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

Using the panels and the commands

You can use the MQSC commands, the PCF commands, or the operations and control panels to manage DQM.

For information about MQSC commands, see [Administering IBM MQ using MQSC commands](#). For information about PCF commands, see [Automating administration using Programmable Command Formats commands](#).

Using the initial panel

For an introduction to invoking the operations and control panels, using the function keys, and getting help, see [Administration de IBM MQ for z/OS](#).

Note: To use the operations and control panels, you must have the correct security authorization; see [Administering IBM MQ for z/OS](#) and sub topics for more information. [Figure 103 on page 1015](#) shows the panel that is displayed when you start a panel session. The text after the panel explains the actions you perform in this panel.

```
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 . . . . . MQ25 - local queue manager or group
Target queue manager . . . MQ25
- connected or remote queue manager for command input
Action queue manager . . . MQ25 - command scope in group
Response wait time . . . . 10 5 - 999 seconds

(C) Copyright IBM Corporation 1993, 2024. All rights reserved.

Command ==>
F1=Help F2=Split F3=Exit F4=Prompt F9=SwapNext F10=Messages
F12=Cancel
```

Figure 103. The operations and controls initial panel

From this panel, you can:

- Select the action you want to perform by typing in the appropriate number in the **Action** field.
- Specify the object type that you want to work with. Press F4 for a list of object types if you are not sure what they are.
- Display a list of objects of the type specified. Type in an asterisk (*) in the **Name** field and press enter to display a list of objects (of the type specified) that have already been defined on this subsystem. You

can then select one or more objects to work with in sequence. [Figure 104 on page 1016](#) shows a list of channels produced in this way.

- Specify the disposition in the queue sharing group of the objects you want to work with in the **Disposition** field. The disposition determines where the object is kept and how the object behaves.
- Choose the local queue manager, or queue sharing group to which you want to connect in the **Connect name** field. If you want the commands to be issued on a remote queue manager, choose either the **Target queue manager** field or the **Action queue manager** field, depending upon whether the remote queue manager is not or is a member of a queue sharing group. If the remote queue manager is not a member of a queue sharing group, choose the **Target queue manager** field. If the remote queue manager is a member of a queue sharing group, choose the **Action queue manager** field.
- Choose the wait time for responses to be received in the **Response wait time** field.

```
List Channels - MQ25                Row 1 of 8

Type action codes, then press Enter. Press F11 to display connection status.
1=Display 2=Define like 3=Alter 4=Manage 5=Perform
6=Start 7=Stop

Name          Type          Disposition  Status
<> *          CHANNEL      ALL          MQ25
- SYSTEM.DEF.CLNTCONN CLNTCONN    QMGR        MQ25
- SYSTEM.DEF.CLUSRCVR CLUSRCVR    QMGR        MQ25 INACTIVE
- SYSTEM.DEF.CLUSSDR  CLUSSDR     QMGR        MQ25 INACTIVE
- SYSTEM.DEF.RECEIVER RECEIVER     QMGR        MQ25 INACTIVE
- SYSTEM.DEF.REQUESTER REQUESTER    QMGR        MQ25 INACTIVE
- SYSTEM.DEF.SENDER   SENDER      QMGR        MQ25 INACTIVE
- SYSTEM.DEF.SERVER   SERVER       QMGR        MQ25 INACTIVE
- SYSTEM.DEF.SVRCONN  SVRCONN     QMGR        MQ25 INACTIVE
***** End of list *****

Command ==>> -----
F1=Help  F2=Split  F3=Exit  F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd   F9=SwapNext F10=Messages F11=Status F12=Cancel
```

Figure 104. Listing channels

Defining a channel on z/OS

On z/OS, you can define a channel by using MQSC commands or using the operations and control panels.

Procedure

- To define a channel using the MQSC commands, use the **DEFINE CHANNEL** command.
- To use the operations and control panels, starting from the initial panel, complete the following fields and press Enter:

Table 66. Operations and control panels: initial panel fields	
Field	Value to enter in field
Action	2 (Define like)
Object type	Channel type (for example SENDER) or CHANNEL
Name	
Disposition	The location of the new object.

You are presented with some panels to complete with information about the name and attributes that you want for the channel you are defining. They are initialized with the default attribute values. Change any that you want to before pressing Enter.

Note: If you entered CHANNEL in the **object type** field, you are presented with the **Select a Valid Channel Type** panel first.

If you want to define a channel with the same attributes as an existing channel, put the name of the channel you want to copy in the **Name** field on the initial panel. The panels are initialized with the attributes of the existing object.

For information about the channel attributes, see [Channel attributes](#).

Note:

1. Name all the channels in your network uniquely. As shown in [Network diagram showing all channels](#), including the source and target queue manager names in the channel name is a good way to do this naming.

What to do next

After you have defined your channel you must secure your channel. For more information, see [“Securing a channel”](#) on page 1018.

Altering a channel definition

You can alter a channel definition using MQSC commands or using the operations and control panels.

To alter a channel definition using the MQSC commands, use ALTER CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	3 (Alter)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.ALTER
Disposition	The location of the stored object.

You are presented with some panels containing information about the current attributes of the channel. Change any of the unprotected fields that you want by over typing the new value, and then press enter to change the channel definition.

For information about the channel attributes, see [Channel attributes](#).

Displaying a channel definition

You can display a channel definition using MQSC commands or using the operations and control panels.

To display a channel definition using the MQSC commands, use DISPLAY CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	1 (List or Display)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.DISPLAY
Disposition	The location of the object.

You are presented with some panels displaying information about the current attributes of the channel.

For information about the channel attributes, see [Channel attributes](#).

Deleting a channel definition

You can delete a channel definition using MQSC commands or using the operations and control panels.

To delete a channel definition using the MQSC commands, use `DELETE CHANNEL`.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	4 (Manage)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.DELETE
Disposition	The location of the object.

You are presented with another panel. Select function type 1 on this panel.

Press enter to delete the channel definition; you are asked to confirm that you want to delete the channel definition by pressing enter again.

Note: The channel initiator has to be running before a channel definition can be deleted (except for client-connection channels).

Displaying information about the channel initiator

You can display information about the channel initiator using MQSC commands or using the operations and control panels.

To display information about the channel initiator using the MQSC commands, use `DISPLAY CHINIT`.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	1 (Display)
Object type	SYSTEM
Name	Blank

You are presented with another panel. Select function type 1 on this panel.

Note:

1. Displaying distributed queuing information might take some time if you have lots of channels.
2. The channel initiator has to be running before you can display information about distributed queuing.

Securing a channel

You can secure a channel using MQSC commands or using the operations and control panels.

To secure a channel using the MQSC commands, use `SET CHLAUTH`.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	8

You are presented with an editor within which you can provide an MQSC command, in this case a `CHLAUTH` command, see [Figure 105 on page 1019](#). When you have finished typing in the command, the

plus signs (+) are needed. Type PF3 to exit from the editor and submit the command to the command server.

```
***** Top of Data *****
000001 SET CHLAUTH(SYSTEM.DEF.SVRCONN) +
000002 TYPE(SSLPEERMAP) +
000003 SSLPEER('CN="John Smith"') +
000004 MCAUSER('PUBLIC')
***** Bottom of Data *****

Command ==>          Scroll ==> PAGE
F1=Help  F3=Exit  F4=LineEdit F12=Cancel
```

Figure 105. Command Entry

The output of the command is then presented to you, see [Figure 106 on page 1019](#)

```
***** Top of Data *****
000001 CSQU000I CSQUTIL IBM MQ for z/OS V7.1.0
000002 CSQU001I CSQUTIL Queue Manager Utility - 2011-04-20 14:42:58
000003 COMMAND TGTQMGR(MQ23) RESPTIME(30)
000004 CSQU127I Executing COMMAND using input from CSQUCMD data set
000005 CSQU120I Connecting to MQ23
000006 CSQU121I Connected to queue manager MQ23
000007 CSQU055I Target queue manager is MQ23
000008 SET CHLAUTH(SYSTEM.DEF.SVRCONN) +
000009 TYPE(SSLPEERMAP) +
000010 SSLPEER('CN="John Smith"') +
000011 MCAUSER('PUBLIC')
000012 CSQN205I COUNT= 2, RETURN=00000000, REASON=00000000
000013 CSQ9022I !MQ23 CSQMCA ' SET CHLAUTH' NORMAL COMPLETION
000014 CSQU057I 1 commands read
000015 CSQU058I 1 commands issued and responses received, 0 failed
000016 CSQU143I 1 COMMAND statements attempted
000017 CSQU144I 1 COMMAND statements executed successfully
000018 CSQU148I CSQUTIL Utility completed, return code=0
Command ==>          Scroll ==> PAGE
F1=Help  F3=Exit  F5=Rfind  F6=Rchange  F9=SwapNext F12=Cancel
```

Figure 106. Command Output

Starting a channel initiator

You can start a channel initiator using MQSC commands or using the operations and control panels.

To start a channel initiator using the MQSC commands, use START CHINIT.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	6 (Start)
Object type	SYSTEM
Name	Blank

The Start a System Function panel is displayed. The text following the following panel explains what action to take:

Start a System Function

Select function type, complete fields, then press Enter to start system function.

```
Function type . . . . . _ 1. Channel initiator
2. Channel listener
Action queue manager . . . : MQ25

Channel initiator
JCL substitution . . . . . -----
-----

Channel listener
Inbound disposition . . . Q G=Group, Q=Qmgr
Transport type . . . . . _ L=LU6.2, T=TCP/IP
LU name (LU6.2) . . . . . -----
Port number (TCP/IP) . . . 1414
IP address (TCP/IP) . . . -----

Command ==> -----
F1=Help  F2=Split  F3=Exit  F9=SwapNext  F10=Messages  F12=Cancel
```

Figure 107. Starting a system function

Select function type 1 (channel initiator), and press enter.

Stopping a channel initiator

You can stop a channel initiator using MQSC commands or using the operations and control panels.

To stop a channel initiator using the MQSC commands, use STOP CHINIT.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	7 (Stop)
Object type	SYSTEM
Name	Blank

The Stop a System Function panel is displayed. The text following the panel explains how you to use this panel:

```

Stop a System Function

Select function type, complete fields, then press Enter to stop system
function.

Function type . . . . . _ 1. Channel initiator
2. Channel listener
Action queue manager . . . : MQ25

Channel initiator
Restart shared channels Y Y=Yes, N=No

Channel listener
Inbound disposition . . . Q G=Group, Q=Qmgr
Transport type . . . . . _ L=LU6.2, T=TCP/IP

Port number (TCP/IP) . . . -----
IP address (TCP/IP) . . . -----

Command ==> -----
F1=Help F2=Split F3=Exit F9=SwapNext F10=Messages F12=Cancel

```

Figure 108. Stopping a function control

Select function type 1 (channel initiator) and press enter.

The channel initiator waits for all running channels to stop in quiesce mode before it stops.

Note: If some of the channels are receiver or requester channels that are running but not active, a stop request issued to either the receiver or sender channel initiator causes it to stop immediately.

However, if messages are flowing, the channel initiator waits for the current batch of messages to complete before it stops.

Starting a channel listener

You can start a channel listener using MQSC commands or using the operations and control panels.

To start a channel listener using the MQSC commands, use START LISTENER.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	6 (Start)
Object type	SYSTEM
Name	Blank

The Start a System Function panel is displayed (see [Figure 107 on page 1020](#)).

Select function type 2 (channel listener). Select Inbound disposition. Select Transport type. If the Transport type is L, select LU name. If the Transport type is T, select Port number and (optionally) IP address. Press enter.

Note: For the TCP/IP listener, you can start multiple combinations of Port and IP address.

Stopping a channel listener

You can stop a channel listener using MQSC commands or using the operations and control panels.

To stop a channel listener using the MQSC commands, use STOP LISTENER.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	7 (Stop)
Object type	SYSTEM
Name	Blank

The Stop a System Function panel is displayed (see [Figure 108 on page 1021](#)).

Select function type 2 (channel listener). Select Inbound disposition. Select Transport type. If the transport type is 'T', select Port number and (optionally) IP address. Press enter.

Note: For a TCP/IP listener, you can stop specific combinations of Port and IP address, or you can stop all combinations.

Starting a channel

You can start a channel using MQSC commands or using the operations and control panels.

To start a channel using the MQSC commands, use START CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	6 (Start)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the object.

The Start a Channel panel is displayed. The text following the panel explains how to use the panel:

```

Start a Channel
Select disposition, then press Enter to start channel.

Channel name . . . . . : CHANNEL.TO.USE
Channel type . . . . . : SENDER
Description . . . . . : Description of CHANNEL.TO.USE

Disposition . . . . . P   P=Private on MQ25
S=Shared on MQ25
A=Shared on any queue manager

Command ==> _____
F1=Help   F2=Split   F3=Exit   F9=SwapNext F10=Messages F12=Cancel

```

Figure 109. Starting a channel

Select the disposition of the channel instance and on which queue manager it is to be started.

Press enter to start the channel.

Starting a shared channel

To start a shared channel, and keep it on a nominated channel initiator, use disposition = S (on the START CHANNEL command, specify CHLDISP(FIXSHARED)).

There can be only one instance of the shared channel running at a time. Attempts to start a second instance of the channel fail.

When you start a channel in this way, the following rules apply to that channel:

- You can stop the channel from any queue manager in the queue sharing group. You can stop it even if the channel initiator on which it was started is not running at the time you issue the stop-channel request. When the channel has stopped, you can restart it by specifying disposition = S (CHLDISP(FIXSHARED)) on the same, or another, channel initiator. You can also start it by specifying disposition = A (CHLDISP(SHARED)).
- If the channel is in the starting or retry state, you can restart it by specifying disposition = S (CHLDISP(FIXSHARED)) on the same or a different channel initiator. You can also start it by specifying disposition = A (CHLDISP(SHARED)).
- The channel is eligible to be trigger started when it goes into the inactive state. Shared channels that are trigger started always have a shared disposition (CHLDISP(SHARED)).
- The channel is eligible to be started with CHLDISP(FIXSHARED), on any channel initiator, when it goes into the inactive state. You can also start it by specifying disposition = A (CHLDISP(SHARED)).
- The channel is not recovered by any other active channel initiator in the queue sharing group when the channel initiator on which it was started is stopped with SHARED(RESTART), or when the channel initiator terminates abnormally. The channel is recovered only when the channel initiator on which it was started is next restarted. This stops failed channel-recovery attempts being passed to other channel initiators in the queue sharing group, which would add to their workload.

Testing a channel

You can test a channel using MQSC commands or using the operations and control panels.

To test a channel using the MQSC commands, use PING CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	5 (Perform)
Object type	SENDER, SERVER, or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the channel object.

The Perform a Channel Function panel is displayed. The text following the panel explains how to use the panel:

Perform a Channel Function

Select function type, complete fields, then press Enter.

```
Function type . . . . . _ 1. Reset 3. Resolve with commit  
2. Ping 4. Resolve with backout
```

```
Channel name . . . . . : CHANNEL.TO.USE  
Channel type . . . . . : SENDER  
Description . . . . . : Description of CHANNEL.TO.USE
```

```
Disposition . . . . . P P=Private on MQ25  
S=Shared on MQ25  
A=Shared on any queue manager
```

```
Sequence number for reset . . 1 1 - 99999999  
Data length for ping . . . . 16 16 - 32768
```

```
Command ==> _____  
F1=Help F2=Split F3=Exit F9=SwapNext F10=Messages F12=Cancel
```

Figure 110. Testing a channel

Select function type 2 (ping).

Select the disposition of the channel for which the test is to be done and on which queue manager it is to be tested.

The data length is initially set to 16. Change it if you want and press enter.

Resetting message sequence numbers for a channel

You can reset message sequence numbers for a channel using MQSC commands or using the operations and control panels.

To reset channel sequence numbers using the MQSC commands, use RESET CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	5 (Perform)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the channel object.

The Perform a Channel Function panel is displayed (see [Figure 110 on page 1024](#)).

Select Function type 1 (reset).

Select the disposition of the channel for which the reset is to be done and on which queue manager it is to be done.

The **sequence number** field is initially set to one. Change this value if you want, and press enter.

Resolving in-doubt messages on a channel

You can resolve in-doubt messages on a channel using MQSC commands or using the operations and control panels.

To resolve in-doubt messages on a channel using the MQSC commands, use RESOLVE CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	5 (Perform)
Object type	SENDER, SERVER, or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the object.

The Perform a Channel Function panel is displayed (see [Figure 110 on page 1024](#)).

Select Function type 3 or 4 (resolve with commit or backout). (See [“Traitement des canaux en attente de validation” on page 257](#) for more information.)

Select the disposition of the channel for which resolution is to be done and which queue manager it is to be done on. Press enter.

Stopping a channel

You can stop a channel using MQSC commands or using the operations and control panels.

To stop a channel using the MQSC commands, use STOP CHANNEL.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	7 (Stop)
Object type	channel type (for example SENDER) or CHANNEL
Name	CHANNEL.TO.USE
Disposition	The disposition of the object.

The Stop a Channel panel is displayed. The text following the panel explains how to use the panel:

Stop a Channel

Complete fields, then press Enter to stop channel.

```
Channel name . . . . . : CHANNEL.TO.USE
Channel type . . . . . : SENDER
Description . . . . . : Description of CHANNEL.TO.USE
```

```
Disposition . . . . . P   P=Private on MQ25
A=Shared on any queue manager
```

```
Stop mode . . . . . 1   1. Quiesce  2. Force
Stop status . . . . . 1   1. Stopped  2. Inactive
```

```
Queue manager . . . . . -----
Connection name . . . . . -----
```

```
Command ==> -----
F1=Help   F2=Split  F3=Exit   F9=SwapNext F10=Messages F12=Cancel
```

Figure 111. Stopping a channel

Select the disposition of the channel for which the stop is to be done and on which queue manager it is to be stopped.

Choose the stop mode that you require:

Quiesce

The channel stops when the current message is completed and the batch is then ended, even if the batch size value has not been reached and there are messages already waiting on the transmission queue. No new batches are started. This mode is the default.

Force

The channel stops immediately. If a batch of messages is in progress, an 'in-doubt' situation can result.

Choose the queue manager and connection name for the channel you want to stop.

Choose the status that you require:

Stopped

The channel is not restarted automatically, and must be restarted manually. This mode is the default if no queue manager or connection name is specified. If a name is specified, it is not allowed.

Inactive

The channel is restarted automatically when required. This mode is the default if a queue manager or connection name is specified.

Press enter to stop the channel.

See “[Arrêt et mise au repos des canaux](#)” on page 255 for more information. For information about restarting stopped channels, see “[Redémarrage des canaux arrêtés](#)” on page 256.

Note: If a shared channel is in a retry state and the channel initiator on which it was started is not running, a STOP request for the channel is issued on the queue manager where the command was entered.

Displaying channel status

You can display channel status by using MQSC commands, or by using the operations and control panels.

To display the status of a channel or a set of channels using the MQSC commands, use DISPLAY CHSTATUS.

Note: Displaying channel status information can take some time if you have lots of channels.

Using the operations and control panels on the List Channel panel (see [Figure 104 on page 1016](#)), a summary of the channel status is shown for each channel as follows:

INACTIVE	No connections are active
<i>status</i>	One connection is active
<i>nnn status</i>	More than one connection is current and all current connections have the same status
<i>nnn CURRENT</i>	More than one connection is current and the current connections do not all have the same status
Blank	IBM MQ is unable to determine how many connections are active (for example, because the channel initiator is not running)

Note: For channel objects with the disposition GROUP, no status is displayed.

where *nnn* is the number of active connections, and *status* is one of the following:

INIT	INITIALIZING
BIND	BINDING
START	STARTING

RUN	RUNNING
STOP	STOPPING or STOPPED
RETRY	RETRYING
REQST	REQUESTING

To display more information about the channel status, press the Status key (F11) on the List Channel or the Display, or Alter channel panels to display the List Channels - Current Status panel (see [Figure 112](#) on page 1027).

```

List Channels - Current Status - MQ25          Row 1 of 16

Type action codes, then press Enter. Press F11 to display saved status.
1=Display current status

Channel name      Connection name      State
Start time      Messages Last message time Type Disposition
<> *
CHANNEL ALL MQ25

- RMA0.CIRCUIT.ACL.F RMA1          STOP
- 2005-03-21 10.22.36 557735 2005-03-24 09.51.11 SENDER PRIVATE MQ25
- RMA0.CIRCUIT.ACL.N RMA1
- 2005-03-21 10.23.09 378675 2005-03-24 09.51.10 SENDER PRIVATE MQ25
- RMA0.CIRCUIT.CL.F RMA2
- 2005-03-24 01.12.51 45544 2005-03-24 09.51.08 SENDER PRIVATE MQ25
- RMA0.CIRCUIT.CL.N RMA2
- 2005-03-24 01.13.55 45560 2005-03-24 09.51.11 SENDER PRIVATE MQ25
- RMA1.CIRCUIT.CL.F RMA1
- 2005-03-21 10.24.12 360757 2005-03-24 09.51.11 RECEIVER PRIVATE MQ25
- RMA1.CIRCUIT.CL.N RMA1
- 2005-03-21 10.23.40 302870 2005-03-24 09.51.09 RECEIVER PRIVATE MQ25
***** End of list *****
Command ==>
F1=Help F2=Split F3=Exit F4=Filter F5=Refresh F7=Bkwd
F8=Fwd F9=SwapNext F10=Messages F11=Saved F12=Cancel

```

Figure 112. Listing channel connections

The values for status are as follows:

INIT	INITIALIZING
BIND	BINDING
START	STARTING
RUN	RUNNING
STOP	STOPPING or STOPPED
RETRY	RETRYING
REQST	REQUESTING
DOUBT	STOPPED and INDOUBT(YES)

See [“Etats des canaux”](#) on page 247 for more information.

You can press F11 to see a similar list of channel connections with saved status; press F11 to get back to the current list. The saved status does not apply until at least one batch of messages has been transmitted on the channel.

Use action code 1 or a slash (/) to select a connection and press enter. The Display Channel Connection Current Status panels are displayed.

Displaying cluster channels

You can display cluster channels using MQSC commands or using the operations and control panels.

To display all the cluster channels that have been defined (explicitly or using auto-definition), use the MQSC command, DISPLAY CLUSQMGR.

Using the operations and control panels, starting from the initial panel, complete these fields and press enter:

Field	Value
Action	1 (List or Display)
Object type	CLUSCHL
Name	*

You are presented with a panel like figure [Figure 113 on page 1028](#), in which the information for each cluster channel occupies three lines, and includes its channel, cluster, and queue manager names. For cluster-sender channels, the overall state is shown.

```
List Cluster queue manager Channels - MQ25      Row 1 of 9
Type action codes, then press Enter. Press F11 to display connection status.
1=Display 5=Perform 6=Start 7=Stop

Channel name      Connection name      State
Type      Cluster name      Suspended
Cluster queue manager name      Disposition
<> *          -      MQ25
- TO.MQ90.T      HURSLEY.MACH90.COM(1590)
- CLUSRCVR      VJH01T              N
- MQ90          -      MQ25
- TO.MQ95.T      HURSLEY.MACH95.COM(1595)      RUN
- CLUSSDRA      VJH01T              N
- MQ95          -      MQ25
- TO.MQ96.T      HURSLEY.MACH96.COM(1596)      RUN
- CLUSSDRB      VJH01T              N
- MQ96          -      MQ25
***** End of list *****

Command ==>-----
F1=Help  F2=Split  F3=Exit  F4=Filter  F5=Refresh  F7=Bkwd
F8=Fwd   F9=SwapNext F10=Messages F11=Status F12=Cancel
```

Figure 113. Listing cluster channels

To display full information about one or more channels, type Action code 1 against their names and press enter. Use Action codes 5, 6, or 7 to perform functions (such as ping, resolve, and reset), and start or stop a cluster channel.

To display more information about the channel status, press the Status key (F11).

Preparing IBM MQ for z/OS to use the zEnterprise Data Compression

Express facility

The zEnterprise® Data Compression (zEDC) Express facility is available for certain models of IBM Z machines, starting from IBM zEC12 GA2, using a minimum z/OS level of z/OS 2.1.

See [zEnterprise Data Compression \(zEDC\)](#) for further information.

Prerequisites

For IBM z15 and later, the zEnterprise Data Compression (zEDC) Express facility was moved from an optional feature in the PCIe I/O drawer of the hardware system to be on-chip as the Integrated

Accelerator for zEDC. With this change, the configuration prerequisites are updated and are dependent on your hardware system.

IBM z15 or later

Apply one of the following PTFs, according to your level of z/OS:

- z/OS 2.4: UJ00636
- z/OS 2.3: UJ00635
- z/OS 2.2: UJ00638
- z/OS 2.1: UJ00639

There are no hardware requirements for z15 or later systems. The Integrated Accelerator for zEDC solution in these systems provides built-in data acceleration, so a separate adapter is no longer required.

IBM zEC12 GA2 to IBM z14

Your system must also have the following requirements:

- A zEDC Express[®] adapter, installed in the PCIe I/O drawers of the hardware system.
- The zEDC software capability (an optional, paid-for feature) must be enabled in an IFAPRDxx parmlib member.

Procedure

IBM zEC12 GA2 to IBM z14

Ensure that the channel initiator user ID has READ authority to the FPZ.ACCELERATOR.COMPRESSION profile in the RACF FACILITY CLASS, or the equivalent in the external security manager (ESM) that your enterprise uses.



Attention: Not required for IBM z15 or later.

IBM zEnterprise zEC12 GA2 or later

Configure the channel with COMPMSG(ZLIBFAST) at both the sending and receiving ends. Once configured, zlib compression is used to compress and decompress messages flowing across the channel.

Compression is performed in the zEDC when the size of the data to be compressed is above the minimum threshold. The threshold is dependent upon the IBM z hardware being used

- IBM zEC12 GA2 to IBM z14 has a minimum threshold of 4KB
- IBM z15 or later has a minimum threshold of 1KB

For messages below the threshold size, compression or inflation is performed in the software.

z/OS Setting up communication for z/OS

When a distributed-queuing management channel is started, it tries to use the connection specified in the channel definition. To succeed, it is necessary for the connection to be defined and available. This section explains how to define a connection.

DQM is a remote queuing facility for IBM MQ. It provides channel control programs for the queue manager that form the interface to communication links. These links are controllable by the system operator. The channel definitions held by distributed queuing management use these connections.

Choose from one of the two forms of communication protocol that can be used for z/OS:

- [“Defining a TCP connection on z/OS” on page 1030](#)
- [“Defining an LU6.2 connection for z/OS using APPC/MVS” on page 1033](#)

Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Each channel definition must specify only one protocol as the transmission protocol (Transport Type) attribute. A queue manager can use more than one protocol to communicate.

You might also find it helpful to refer to [Example configuration - IBM MQ for z/OS](#) . If you are using queue sharing groups, see [“Setting up communication for IBM MQ for z/OS using queue sharing groups”](#) on page 1038.

Related concepts

[“Using the panels and the commands”](#) on page 1015

You can use the MQSC commands, the PCF commands, or the operations and control panels to manage DQM.

[“Setting up IBM MQ for z/OS”](#) on page 935

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

[“Monitoring and controlling channels on z/OS”](#) on page 1013

Use the DQM commands and panels to create, monitor, and control the channels to remote queue managers.

[“Preparing IBM MQ for z/OS for DQM with queue sharing groups”](#) on page 1034

Use the instructions in this section to configure distributed queuing with queue sharing groups on IBM MQ for z/OS.

[“Setting up communication for IBM MQ for z/OS using queue sharing groups”](#) on page 1038

When a distributed-queuing management channel is started, it attempts to use the connection specified in the channel definition. For this attempt to succeed, it is necessary for the connection to be defined and available.

Related tasks

[“Configuration de la file d'attente répartie”](#) on page 212

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ , y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

[“Setting up communications with other queue managers on z/OS”](#) on page 1009

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

z/OS *Defining a TCP connection on z/OS*

To define a TCP connection, there are a number of settings to configure.

The TCP address space name must be specified in the TCP system parameters data set, *tcPIP.TCPIP.DATA*. In the data set, a "TCPIPJOBNAME *TCPIP_proc*" statement must be included.

If you are using a firewall, you need to configure allow connections from the channel initiator to the addresses in the channels, and from the remote connections into the queue manager.

Typically the definition for a firewall configures the sending IP address and port to the destination IP address and port:

- A z/OS image can have more than one host name, and you might need to configure the firewall with multiple host addresses as the source address.

You can use the NETSTAT HOME command to display these names and addresses.

- A channel initiator can have multiple listeners on different ports, so you need to configure these ports.
- If you are using a shared port for a queue sharing group you must configure the shared port as well.

The channel initiator address space must have authority to read the data set. The following techniques can be used to access your TCPIP.DATA data set, depending on which TCP/IP product and interface you are using:

- Environment variable, RESOLVER_CONFIG
- /etc/resolv.conf on the file system
- //SYSTCPD DD statement
- //SYSTCPDD DD statement
- *jobname/userid*.TCPIP.DATA
- SYS1.TCPPARMS(TCPDATA)
- *zapname*.TCPIP.DATA

You must also be careful to specify the high-level qualifier for TCP/IP correctly.

You need a suitably configured Domain Name System (DNS) server, capable of both Name to IP address translation and IP address to Name translation.

Note: Some changes to the resolver configuration require a recycle of applications using it, for example, IBM MQ.

For more information, see the following:

- [Base TCP/IP system](#)
- [z/OS UNIX System Services](#).

Each TCP channel when started uses TCP resources; you might need to adjust the following parameters in your PROFILE.TCPIP configuration data set:

ACBPOOLSIZE

Add one per started TCP channel, plus one

CCBPOOLSIZE

Add one per started TCP channel, plus one per DQM dispatcher, plus one

DATABUFFERPOOLSIZE


Add two per started TCP channel, plus one

MAXFILEPROC

Controls how many channels each dispatcher in the channel initiator can handle.

This parameter is specified in the BPXPRMxx member of SYSI.PARMLIB. Ensure that you specify a value large enough for your needs.

By default, the channel initiator is only capable of binding to IP addresses associated with the stack named in the TCPNAME queue manager attribute. To allow the channel initiator to communicate using additional TCP/IP stacks on the system, change the TCPSTACK queue manager attribute to MULTIPLE.

 Un canal de transmission de messages qui utilise TCP/IP peut pointer vers un IBM Aspera faspio Gateway, qui fournit un tunnel TCP/IP rapide pouvant augmenter considérablement le débit du réseau. Un gestionnaire de files d'attente s'exécutant sur une plateforme autorisée peut se connecter via un Aspera gateway. La passerelle elle-même est déployée sur Red Hat , Ubuntu Linux ou Windows. Voir [Définition d'une connexion Aspera gateway sous Linux ou Windows](#).

Related concepts

[“Sending end” on page 1032](#)

At the sending end of the TCP/IP connection, there are a number of settings to configure.

[“Receiving on TCP” on page 1032](#)

At the receiving end of the TCP/IP connection, there are a number of settings to configure.

[“Using the TCP listener backlog option on z/OS” on page 1032](#)

When receiving on TCP/IP, a maximum number of outstanding connection requests is set. These outstanding requests can be considered a *backlog* of requests waiting on the TCP/IP port for the listener to accept the request.

Sending end

At the sending end of the TCP/IP connection, there are a number of settings to configure.

The connection name (CONNNAME) field in the channel definition must be set to either the host name (for example MVSHUR1) or the TCP network address of the target. The TCP network address can be in IPv4 dotted decimal form (for example 127.0.0.1) or IPv6 hexadecimal form (for example 2001:DB8:0:0:0:0:0:0). If the connection name is a host name, a TCP name server is required to convert the host name into a TCP host address. (This requirement is a function of TCP, not IBM MQ.)

On the initiating end of a connection (sender, requester, and server channel types) it is possible to provide an optional port number for the connection, for example:

Connection name

192.0.2.0(1555)

In this case the initiating end attempts to connect to a receiving program listening on port 1555.

Note: The default port number of 1414 is used if an optional port number is not specified.

The channel initiator can use any TCP/IP stack which is active and available. By default, the channel initiator binds its outbound channels to the default IP address for the TCP/IP stack named in the TCPNAME queue manager attribute. To connect through a different stack, you need to specify either the host name or IP address of the stack in the LOCLADDR attribute of the channel.

Receiving on TCP

At the receiving end of the TCP/IP connection, there are a number of settings to configure.

Receiving channel programs are started in response to a startup request from the sending channel. To do so, a listener program has to be started to detect incoming network requests and start the associated channel. You start this listener program with the [START LISTENER](#) command, or using the operations and control panels.

By default:

- The TCP Listener program uses port 1414 and listens on all addresses available to your TCP stack.
- TCP/IP listeners can bind only to addresses associated with the TCP/IP stack named in the TCPNAME queue manager attribute.

To start listeners for other addresses, or all available TCP stacks, set your TCPSTACK queue manager attribute to 'MULTIPLE'.

You can start your TCP listener program to listen only on a specific address or host name by specifying IPADDR in the START LISTENER command. For more information, see [Listeners](#).

Using the TCP listener backlog option on z/OS

When receiving on TCP/IP, a maximum number of outstanding connection requests is set. These outstanding requests can be considered a *backlog* of requests waiting on the TCP/IP port for the listener to accept the request.

The default listener backlog value on z/OS is 10000. If the backlog reaches this values, the TCP/IP connection is rejected and the channel is not able to start.

For MCA channels, this results in the channel going into a RETRY state and retrying the connection at a later time.

For client connections, the client receives an MQRC_Q_MGR_NOT_AVAILABLE reason code from MQCONN and can retry the connection at a later time.

Related concepts

[“Utilisation de l'option Journal des éléments en attente du programme d'écoute TCP sous IBM MQ for Multiplatforms” on page 290](#)

Dans TCP, les connexions sont traitées comme incomplètes sauf si un établissement de liaison à trois est effectué entre le serveur et le client. Ces connexions sont appelées demandes de connexion en attente. Une valeur maximale est définie pour ces demandes de connexion en attente et peut être considérée

comme un journal des demandes en attente sur le port TCP pour que le programme d'écoute accepte la demande.

Defining an LU6.2 connection for z/OS using APPC/MVS

To define an LU6.2 connection there are a number of settings to configure.

APPC/MVS setup

Each instance of the channel initiator must have the name of the LU that it is to use defined to APPC/MVS, in the APPCPMxx member of SYS1.PARMLIB, as in the following example:

```
LUADD ACBNAME( luname ) NOSCHED TPDATA(CSQ.APPCTP)
```

luname is the name of the logical unit to be used. NOSCHED is required; TPDATA is not used. No additions are necessary to the ASCHPMxx member, or to the APPC/MVS TP profile data set.

The side information data set must be extended to define the connections used by DQM. See the supplied sample CSQ4SIDE for details of how to do this using the APPC utility program ATBSDFMU. For details of the TPNAME values to use, see the following table for information:

Remote platform	TPNAME
z/OS or MVS	The same as TPNAME in the corresponding side information about the remote queue manager.
IBM i	The same as the compare value in the routing entry on the IBM i system.
AIX and Linux systems	The same as TPNAME in the corresponding side information about the remote queue manager.
Windows	As specified in the Windows Run Listener command, or the invocable Transaction Program that was defined using TpSetup on Windows.

If you have more than one queue manager on the same machine, ensure that the TPnames in the channel definitions are unique.

In an environment where the queue manager is communicating using APPC with a queue manager on the same or another z/OS system, ensure that either the VTAM definition for the communicating LU specifies SECACPT(ALREADYV), or that there is a RACF APPCLU profile for the connection between LUs, which specifies CONVSEC(ALREADYV).

The z/OS command VARY ACTIVE must be issued against both base and listener LUs before attempting to start either inbound or outbound communications.



Attention: In addition to the APPC setup, you must issue the following command:

```
ALTER QMGR LUNAME(luname)
```

and restart the channel initiator.

See [LUNAME](#) for further information.

Related concepts

[“Connecting to LU 6.2” on page 1033](#)

To connect to LU 6.2, there are a number of settings to configure.

[“Receiving on LU 6.2” on page 1034](#)

To receive on LU 6.2, there are a number of settings to configure.

Connecting to LU 6.2

To connect to LU 6.2, there are a number of settings to configure.

The connection name (CONNNAME) field in the channel definition must be set to the symbolic destination name, as specified in the side information data set for APPC/MVS.

The LU name to use (defined to APPC/MVS as described previously) must also be specified in the channel initiator parameters. It must be set to the same LU that is used for receiving by the listener.

The channel initiator uses the "SECURITY(SAME)" APPC/MVS option, so it is the user ID of the channel initiator address space that is used for outbound transmissions, and is presented to the receiver.

Receiving on LU 6.2

To receive on LU 6.2, there are a number of settings to configure.

Receiving MCAs are started in response to a startup request from the sending channel. To do so, a listener program has to be started to detect incoming network requests and start the associated channel. The listener program is an APPC/MVS server. You start it with the START LISTENER command, or using the operations and control panels. You must specify the LU name to use with a symbolic destination name defined in the side information data set. The local LU so identified must be the same as the one used for outbound transmissions, as set in the channel initiator parameters.

Preparing IBM MQ for z/OS for DQM with queue sharing groups

Use the instructions in this section to configure distributed queuing with queue sharing groups on IBM MQ for z/OS.

For an example configuration using queue sharing groups, see [Example configuration - IBM MQ for z/OS using queue sharing groups](#). For a message channel planning example using queue sharing groups, see [Message channel planning example for z/OS using queue sharing groups](#).

You need to create and configure the following components to enable distributed queuing with queue sharing groups:

- [LU 6.2 and TCP/IP listeners](#)
- [Transmission queues and triggering](#)
- [Message channel agents](#)
- [Synchronization queue](#)

After you have created the components you need to set up the communication, see [“Setting up communication for IBM MQ for z/OS using queue sharing groups” on page 1038](#).

For information about how to monitor and control channels when using queue sharing groups, see [“Monitoring and controlling channels on z/OS” on page 1013](#).

See the following sections for queue sharing group concepts and benefits.

Class of service

A shared queue is a type of local queue that offers a different class of service. Messages on a shared queue are stored in a coupling facility (CF), which allows them to be accessed by all queue managers in the queue sharing group. A message on a shared queue must be a message of length no more than 100 MB.

Generic interface

A queue sharing group has a generic interface that allows the network to view the group as a single entity. This view is achieved by having a single generic address that can be used to connect to any queue manager within the group.

Each queue manager in the queue sharing group listens for inbound session requests on an address that is logically related to the generic address. For more information see [“LU 6.2 and TCP/IP listeners for queue sharing groups” on page 1036](#).

Load-balanced channel start

A shared transmission queue can be serviced by an outbound channel running on any channel initiator in the queue sharing group. Load-balanced channel start determines where a start channel command is targeted. An appropriate channel initiator is chosen that has access to the necessary communications subsystem. For example, a channel defined with TRPTYPE(LU6.2) cannot be started on a channel initiator that only has access to a TCP/IP subsystem.

The choice of channel initiator is dependent on the channel load and the headroom of the channel initiator. The channel load is the number of active channels as a percentage of the maximum number of active channels allowed as defined in the channel initiator parameters. The headroom is the difference between the number of active channels and the maximum number allowed.

Inbound shared channels can be load-balanced across the queue sharing group by use of a generic address, as described in [“LU 6.2 and TCP/IP listeners for queue sharing groups”](#) on page 1036.

Shared channel recovery

The following table shows the types of shared-channel failure and how each type is handled.

Type of failure:	What happens:
Channel initiator communications subsystem failure	The channels dependent on the communications subsystem enter channel retry, and are restarted on an appropriate queue sharing group channel initiator by a load-balanced start command.
Channel initiator failure	The channel initiator fails, but the associated queue manager remains active. The queue manager monitors the failure and initiates recovery processing.
Queue manager failure	The queue manager fails (failing the associated channel initiator). Other queue managers in the queue sharing group monitor the event and initiate peer recovery.
Shared status failure	Channel state information is stored in Db2, so a loss of connectivity to Db2 becomes a failure when a channel state change occurs. Running channels can carry on running without access to these resources. On a failed access to Db2, the channel enters retry.

Shared channel recovery processing on behalf of a failed system requires connectivity to Db2 to be available on the system managing the recovery to retrieve the shared channel status.

Client channels

Client connection channels can benefit from the high availability of messages in queue sharing groups that are connected to the generic interface instead of being connected to a specific queue manager. For more information, see [Client connection channels](#).

Related concepts

[Shared queues and queue sharing groups](#)

[“Setting up IBM MQ for z/OS”](#) on page 935

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

[“Clusters and queue sharing groups”](#) on page 1037

You can make your shared queue available to a cluster in a single definition. To do so you specify the name of the cluster when you define the shared queue.

[“Channels and serialization”](#) on page 1038

During shared queue peer recovery, message channel agents that process messages on shared queues serialize their access to the queues.

[Intra-group queuing](#)

Related tasks

[“Configuration de la file d'attente répartie” on page 212](#)

Cette section fournit des informations plus détaillées sur l'intercommunication entre les installations IBM MQ , y compris la définition de file d'attente, la définition de canal, le déclenchement et les procédures de point de synchronisation.

[“Setting up communications with other queue managers on z/OS” on page 1009](#)

This section describes the IBM MQ for z/OS preparations you need to make before you can start to use distributed queuing.

LU 6.2 and TCP/IP listeners for queue sharing groups

The group LU 6.2 and TCP/IP listeners listen on an address that is logically connected to the generic address.

For the LU 6.2 listener, the specified LUGROUP is mapped to the VTAM generic resource associated with the queue sharing group. For an example of setting up this technology, see [“Defining an LU6.2 connection for z/OS using APPC/MVS” on page 1033](#).

For the TCP/IP listener, the specified port can be connected to the generic address in one of the following ways:

- For a front-end router such as the IBM Network Dispatcher, inbound connect requests are forwarded from the router to the members of the queue sharing group.
- For TCP/IP Sysplex Distributor, each listener that is running and is listening on a particular address that is set up as a Distributed DVIPA is allocated a proportion of the incoming requests. For an example of setting up this technology, see [Using Sysplex Distributor](#)

Transmission queues and triggering for queue sharing groups

A shared transmission queue is used to store messages before they are moved from the queue sharing group to the destination.

It is a shared queue and it is accessible to all queue managers in the queue sharing group.

Triggering

A triggered shared queue can generate more than one trigger message for a satisfied trigger condition. There is one trigger message generated for each local initiation queue defined on a queue manager in the queue sharing group associated with the triggered shared queue.

For distributed queuing, each channel initiator receives a trigger message for a satisfied shared transmission queue trigger condition. However, only one channel initiator actually processes the triggered start, and the others fail safely. The triggered channel is then started with a load balanced start (see [“Preparing IBM MQ for z/OS for DQM with queue sharing groups” on page 1034](#)) that is triggered to start channel QSG . TO . QM2. To create a shared transmission queue, use the IBM MQ commands (MQSC) as shown in the following example:

```
DEFINE QLOCAL(QM2) DESCR('Transmission queue to QM2') +  
USAGE(XMITQ) QSGDISP(SHARED) +  
CFSTRUCT(APPLICATION1) INITQ(SYSTEM.CHANNEL.INITQ) +  
TRIGGER TRIGDATA(QSG.TO.QM2)
```

Note: If a shared queue is setup for triggering and connection to the Coupling Facility hosting the shared queue is lost, a trigger event might be generated and a message put to the initiation queue. This can happen even when no message was put to the original shared queue setup for triggering. This is caused by the over-indication of bits by the IXLVECTR macro as documented in [The List Notification Vector](#).

Message channel agents for queue sharing groups

A channel can only be started on a channel initiator if it has access to a channel definition for a channel with that name.

A message channel agent is an IBM MQ program that controls the sending and receiving of messages. Message channel agents move messages from one queue manager to another; there is one message channel agent at each end of a channel.

A channel definition can be defined to be private to a queue manager or stored on the shared repository and available anywhere (a group definition). This means that a group defined channel is available on any channel initiator in the queue sharing group.

Note: The private copy of the group definition can be changed or deleted.

To create group channel definitions, use the IBM MQ commands (MQSC) as shown in the following examples:

```
DEFINE CHL(QSG.TO.QM2) CHLTYPE(SDR) +  
TRPTYPE(TCP) CONNAME(QM2.MACH.IBM.COM) +  
XMITQ(QM2) QSGDISP(GROUP)
```

```
DEFINE CHL(QM2.TO.QSG) CHLTYPE(RCVR) TRPTYPE(TCP) +  
QSGDISP(GROUP)
```

There are two perspectives from which to look at the message channel agents used for distributed queuing with queue sharing groups:

Inbound

An inbound channel is a shared channel if it is connected to the queue manager through the group listener. It is connected either through the generic interface to the queue sharing group, then directed to a queue manager within the group, or targeted at the group port of a specific queue manager or the luname used by the group listener.

Outbound

An outbound channel is a shared channel if it moves messages from a shared transmission queue. In the example commands, sender channel QSG.TO.QM2 is a shared channel because its transmission queue, QM2 is defined with QSGDISP(SHARED).

Synchronization queue for queue sharing groups

Shared channels have their own shared synchronization queue called SYSTEM.QSG.CHANNEL.SYNCQ.

This synchronization queue is accessible to any member of the queue sharing group. (Private channels continue to use the private synchronization queue. See [“Defining IBM MQ objects on z/OS” on page 1012](#)). This means that the channel can be restarted on a different queue manager and channel initiator instance within the queue sharing group in the event of failure of the communications subsystem, channel initiator, or queue manager. For further information, see [“Preparing IBM MQ for z/OS for DQM with queue sharing groups” on page 1034](#).

DQM with queue sharing groups requires that a shared queue is available with the name SYSTEM.QSG.CHANNEL.SYNCQ. This queue must be available so that a group listener can successfully start.

If a group listener fails because the queue was not available, the queue can be defined and the listener can be restarted without recycling the channel initiator. The non-shared channels are not affected.

Make sure that you define this queue using INDXTYPE(MSGID). This definition improves the speed at which messages on the queue can be accessed.

Clusters and queue sharing groups

You can make your shared queue available to a cluster in a single definition. To do so you specify the name of the cluster when you define the shared queue.

Users in the network see the shared queue as being hosted by each queue manager within the queue sharing group. (The shared queue is not advertised as being hosted by the queue sharing group). Clients can start sessions with all members of the queue sharing group to put messages to the same shared queue.

For more information, see [“Configuration d'un cluster de gestionnaires de files d'attente”](#) on page 316.

Channels and serialization

During shared queue peer recovery, message channel agents that process messages on shared queues serialize their access to the queues.

If a queue manager in a queue sharing group fails while a message channel agent is dealing with uncommitted messages on one or more shared queues, the channel and the associated channel initiator will end, and shared queue peer recovery will take place for the queue manager.

Because shared queue peer recovery is an asynchronous activity, peer channel recovery might try to simultaneously restart the channel in another part of the queue sharing group before shared queue peer recovery is complete. If this event happens, committed messages might be processed ahead of the messages still being recovered. To ensure that messages are not processed out of sequence in this way, message channel agents that process messages on shared queues serialize their access to these queues.

An attempt to start a channel for which shared queue peer recovery is still in progress might result in a failure. An error message indicating that recovery is in progress is issued, and the channel is put into retry state. Once queue manager peer recovery is complete, the channel can restart at the time of the next retry.

An attempt to RESOLVE, PING, or DELETE a channel can fail for the same reason.

Setting up communication for IBM MQ for z/OS using queue sharing groups

When a distributed-queuing management channel is started, it attempts to use the connection specified in the channel definition. For this attempt to succeed, it is necessary for the connection to be defined and available.

Choose from one of the two forms of communication protocol that can be used:

- [TCP](#)
- [LU 6.2 through APPC/MVS](#)

You might find it useful to refer to [Example configuration - IBM MQ for z/OS using queue sharing groups](#).

Defining a TCP connection for queue sharing groups

To define a TCP connection for a queue sharing group, certain attributes on the sending and receiving end must be configured.

For information about setting up your TCP, see [“Defining a TCP connection on z/OS”](#) on page 1030.

Sending end

The connection name (CONNNAME) field in the channel definition to connect to your queue sharing group must be set to the generic interface of your queue sharing group (see [Queue sharing groups](#)). For more details, refer to [Using Sysplex Distributor](#).

Receiving on TCP using a queue sharing group

Receiving shared channel programs are started in response to a startup request from the sending channel. To do so, a listener must be started to detect incoming network requests and start the associated channel. You start this listener program with the START LISTENER command, using the inbound disposition of the group, or using the operations and control panels.

All group listeners in the queue sharing group must be listening on the same port. If you have more than one channel initiator running on a single MVS image you can define virtual IP addresses and start your

TCP listener program to only listen on a specific address or host name by specifying IPADDR in the START LISTENER command. (For more information, see [START LISTENER](#).)

z/OS Defining an LU 6.2 connection on z/OS

To define an LU 6.2 connection for a queue sharing group, certain attributes on the sending and receiving end must be configured.

For information about setting up APPC/MVS, see [Setting up communication for z/OS](#).

Connecting to APPC/MVS (LU 6.2)

The connection name (CONNNAME) field in the channel definition to connect to your queue sharing group must be set to the symbolic destination name, as specified in the side information data set for APPC/MVS. The partner LU specified in this symbolic destination must be the generic resource name. For more details, see [Defining yourself to the network using generic resources](#).

Receiving on LU 6.2 using a generic interface

Receiving shared MCAs are started in response to a startup request from the sending channel. To do so, a group listener program must be started to detect incoming network requests and start the associated channel. The listener program is an APPC/MVS server. You start it with the START LISTENER command, using an inbound disposition group, or using the operations and control panels. You must specify the LU name to use a symbolic destination name defined in the side information data set. For more details, see [Defining yourself to the network using generic resources](#).

z/OS Using IBM MQ with IMS

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

To configure IBM MQ and IMS to work together, you must complete the following tasks:

- [“Setting up the IMS adapter” on page 1039](#)
- [“Setting up the IMS bridge” on page 1046](#)

Related concepts

[IBM MQ and IMS](#)

[“Using IBM MQ with CICS” on page 1047](#)

To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

[“Using OTMA exits in IMS” on page 1049](#)

Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

[IMS and IMS bridge applications on IBM MQ for z/OS](#)

Related tasks

[“Configuring queue managers on z/OS” on page 930](#)

Use these instructions to configure queue managers on IBM MQ for z/OS.

Related reference

[“Upgrading and applying service to Language Environment or z/OS Callable Services” on page 1047](#)

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS Setting up the IMS adapter

To use IBM MQ within IMS requires the IBM MQ - IMS adapter (generally referred to as the IMS adapter).

This topic tells you how to make the IMS adapter available to your IMS subsystem. If you are not familiar with tailoring an IMS subsystem, see the [IMS documentation](#).

To make the IMS adapter available to IMS applications, follow these steps:

1. Define IBM MQ to IMS as an external subsystem using the IMS external subsystem attach facility (ESAF).

See [“Defining IBM MQ to IMS” on page 1041](#).

2. Include the IBM MQ load library thlqual.SCSQAUTH in the JOBLIB or STEPLIB concatenation in the JCL for your IMS control region and for any dependent region that connects to IBM MQ (if it is not in the LPA or link list). If your JOBLIB or STEPLIB is not authorized, also include it in the DFSESL concatenation after the library containing the IMS modules (usually IMS RESLIB).

Also include thlqual.SCSQANLx (where x is the language letter).

If DFSESL is present, then SCSQAUTH and SCSQANLx need to be included in the concatenation or added to LNKLIST. Adding to the STEPLIB or JOBLIB concatenation in the JCL is not sufficient.

3. Copy the IBM MQ assembler program CSQQDEFV from thlqual.SCSQASMS to a user library.
4. The supplied program, CSQQDEFV, contains one subsystem name CSQ1 identified as default with an IMS language interface token (LIT) of MQM1. You can retain this name for testing and installation verification.

For production subsystems, you change the NAME=CSQ1 to your own subsystem name, or use CSQ1. You can add further subsystem definitions as required. See [“Defining IBM MQ queue managers to the IMS adapter” on page 1044](#) for further information on LITs.

5. Assemble and link-edit the program to produce the CSQQDEFV load module. For the assembly, include the library thlqual.SCSQMACS in your SYSLIB concatenation; use the link-edit parameter RENT. This is shown in the sample JCL in thlqual.SCSQPROC(CSQ4DEFV).
6. Include the user library containing the module CSQQDEFV that you created in the JOBLIB or STEPLIB concatenation in the JCL for any dependent region that connects to IBM MQ. Put this library before the SCSQAUTH because SCSQAUTH has a default load module. If you do not do this, you will receive a user 3041 abend from IMS.
7. If the IMS adapter detects an unexpected IBM MQ error, it issues a z/OS SNAP dump to DD name CSQSNAP and issues reason code MQRC_UNEXPECTED_ERROR to the application. If the CSQSNAP DD statement was not in the IMS dependent region JCL, no dump is taken. If this happens, you could include the CSQSNAP DD statement in the JCL and rerun the application. However, because some problems might be intermittent, it is recommended that you include the CSQSNAP DD statement to capture the reason for failure at the time it occurs.
8. If you want to use dynamic IBM MQ calls (described in [Dynamically calling the IBM MQ stub](#)), build the dynamic stub, as shown in [Figure 114 on page 1041](#).
9. If you want to use the IMS trigger monitor, define the IMS trigger monitor application CSQQTRMN, and perform PSBGEN and ACBGEN. See [“Setting up the IMS trigger monitor” on page 1045](#).
10. If you are using RACF to protect resources in the OPERCMDS class, ensure that the userid associated with your IBM MQ queue manager address space has authority to issue the MODIFY command to any IMS system to which it might connect.


```

//DYNSTUB EXEC PGM=IEWL,PARM='RENT,REUS,MAP,XREF'
//SYSPRINT DD SYSOUT=*
//ACSQMOD DD DISP=SHR,DSN=thlqual.SCSQLOAD
//IMSLIB DD DISP=SHR,DSN=ims.reslib
//SYSLMOD DD DISP=SHR,DSN=private.load1
//SYSUT1 DD UNIT=SYSDA,SPACE=(CYL,1)
//SYSLIN DD *
INCLUDE ACSQMOD(CSQSTUB)
INCLUDE IMSLIB(DFSLI000)
ALIAS MQCONN,MQCONN,MQDISC MQI entry points
ALIAS MQGET,MQPUT,MQPUT1 MQI entry points
ALIAS MQOPEN,MQCLOSE MQI entry points
ALIAS MQBACK,MQCMIT MQI entry points
ALIAS CSQBBAK,CSQBCMT MQI entry points
ALIAS MQINQ,MQSET MQI entry points
ALIAS DFSPLI,PLITDLI IMS entry points
ALIAS DFSCOBOL,CBLTDLI IMS entry points
ALIAS DFSFOR,FORTDLI IMS entry points
ALIAS DFSASM,ASMTDLI IMS entry points
ALIAS DFSPASCL,PASTDLI IMS entry points
ALIAS DFHEI01,DFHEI1 IMS entry points
ALIAS DFSAIBLI,AIBTDLI IMS entry points
ALIAS DFSESS,DSNWLI,DSNHLI IMS entry points
ALIAS MQCRTMH,MQDLTMH,MQDLTMP IMS entry points
ALIAS MQINQMP,MQSETMP,MQMHBUF,MQBUFMH IMS entry points
MODE AMODE(31),RMODE(24) Note RMODE setting
NAME CSQDYNS(R)
/*

```

¹Specify the name of a library accessible to IMS applications that want to make dynamic calls to IBM MQ.

Figure 114. Sample JCL to link-edit the dynamic call stub

Related concepts

IBM MQ and IMS

[“Setting up the IMS bridge” on page 1046](#)

The IBM MQ - IMS bridge is an optional component that enables IBM MQ to input and output to and from existing programs and transactions that are not IBM MQ-enabled.

[IMS and IMS bridge applications on IBM MQ for z/OS](#)

Defining IBM MQ to IMS

IBM MQ must be defined to the IMS control region, and to each dependent region accessing that IBM MQ queue manager. To do this, you must create a subsystem member (SSM) in the IMS.PROCLIB library, and identify the SSM to the applicable IMS regions.

Placing the subsystem member entry in IMS.PROCLIB

Each SSM entry in IMS.PROCLIB defines a connection from an IMS region to a different queue manager.

To name an SSM, concatenate the value (one to four alphanumeric characters) of the IMSID field of the IMS IMCTRL macro with any name (one to four alphanumeric characters) defined by your site.

One SSM can be shared by all the IMS regions, or a specific member can be defined for each region.

This member contains as many entries as there are connections to external subsystems. Each entry is an 80-character record.

Positional parameters

The fields in this entry are:

```
SSN,LIT,ESMT,RTT,REO,CRC
```

where:

SSN

Specifies the IBM MQ queue manager name. It is required, and must contain one through four characters.

LIT

Specifies the language interface token (LIT) supplied to IMS. This field is required, its value must match one in the CSQQDEFV module.

ESMT

Specifies the external subsystem module table (ESMT). This table specifies which attachment modules must be loaded by IMS. CSQQESMT is the required value for this field.

RTT

This option is not supported by IBM MQ.

REO

Specifies the region error option (REO) to be used if an IMS application references a non-operational external subsystem or if resources are unavailable at create thread time. This field is optional and contains a single character, which can be:

R

Passes a return code to the application, indicating that the request for IBM MQ services failed.

Q

Ends the application with an abend code U3051, backs out activity to the last commit point, does a PSTOP of the transaction, and requeues the input message. This option only applies when an IMS application tries to reference a non-operational external subsystem or if the resources are unavailable at create thread time.

IBM MQ completion and reason codes are returned to the application if the IBM MQ problem occurs while IBM MQ is processing the request; that is, after the adapter has passed the request on to IBM MQ.

A

Ends the application with an abend code of U3047 and discards the input message. This option only applies when an IMS application references a non-operational external subsystem or if the resources are unavailable at create thread time.

IBM MQ completion and reason codes are returned to the application if the IBM MQ problem occurs while IBM MQ is processing the request; that is, after the adapter has passed the request on to IBM MQ.

CRC

This option can be specified but is not used by IBM MQ.

Note: For full details of all positional parameters refer to [How external subsystems are specified to IMS](#).

An example SSM entry is:

```
CSQ1,MQM1,CSQQESMT,,R,
```

where:

CSQ1

The default subsystem name as supplied with IBM MQ. You can change this to suit your installation.

MQM1	The default LIT as supplied in CSQQDEFV.
CSQQESMT	The external subsystem module name. You must use this value.
R	REO option.

Keyword parameters

IBM MQ parameters can be specified in keyword format. The SST parameter can have a value of either DB2 or MQ. Support for the MQ value was added in IMS 14. Use of MQ aids clarity, and the IMS subsystem command now includes the SST value, but otherwise does not have any significant effect. A value of DB2 can still be used if required. Other parameters are as described in [Positional parameters](#), and shown in the following example:

```
SST=MQ ,SSN=SYS3 ,LIT=MQM3 ,ESMT=CSQQESMT
```

where:

SYS3	The subsystem name
MQM3	The LIT as supplied in CSQQDEFV
CSQQESMT	The external subsystem module name

Specifying the SSM EXEC parameter

Specify the SSM EXEC parameter in the startup procedure of the IMS control region. This parameter specifies the one-character to four-character subsystem member name (SSM).

If you specify the SSM for the IMS control region, any dependent region running under the control region can attach to the IBM MQ queue manager named in the IMS.PROCLIB member specified by the SSM parameter. The IMS.PROCLIB member name is the IMS ID (IMSID= *xxxx*) concatenated with the one to four characters specified in the SSM EXEC parameter. The IMS ID is the IMSID parameter of the IMCTRL generation macro.

IMS lets you define as many external subsystem connections as are required. More than one connection can be defined for different IBM MQ queue managers. All IBM MQ connections must be within the same z/OS system. For a dependent region, you can specify a dependent region SSM or use the one specified for the control region. You can specify different region error options (REOs) in the dependent region SSM and the control region SSM. [Table 68 on page 1043](#) shows the different possibilities of SSM specifications.

SSM for control region	SSM for dependent region	Action	Comments
No	No	None	No external subsystem can be connected.
No	Yes	None	No external subsystem can be connected.
Yes	No	Use the control region SSM	Applications scheduled in the region can access external subsystems identified in the control region SSM. Exits and control blocks for each attachment are loaded into the control region and the dependent region address spaces.
Yes	Yes (empty)	No SSM is used for the dependent region	Applications scheduled in this region can access DL/I databases only. Exits and control blocks for each attachment are loaded into the control region address space.

Table 68. SSM specifications options (continued)

SSM for control region	SSM for dependent region	Action	Comments
Yes	Yes (not empty)	Check the dependent region SSM with the control region SSM	Applications scheduled in this region can access only external subsystems identified in both SSMs. Exits and control blocks for each attachment are loaded into the control region and the dependent region address spaces.

There is no specific parameter to control the maximum number of SSM specification possibilities.

Preloading the IMS adapter

The performance of the IMS adapter can be improved if it is preloaded by IMS. Preloading is controlled by the DFSMPLxx member of IMS.PROCLIB: see "IMS Administration Guide: System" for more information. The IBM MQ module names to specify are:

CSQACLST	CSQAMLST	CSQAPRH	CSQAVICM	CSQFSALM	CSQQDEFV
CSQQCONN	CSQQDISC	CSQQTERM	CSQQINIT	CSQQBACK	CSQQCMMT
CSQQESMT	CSQQPREP	CSQQTTHD	CSQQWAIT	CSQQNORM	CSQQSSOF
CSQQSSON	CSQFSTAB	CSQQRESV	CSQQSNOP	CSQQCMND	CSQQCVER
CSQQTMID	CSQQTRGI	CSQQCON2	CSQBPAPI	CSQBCRMH	CSQBAPPL

For more information on the use of IBM MQ classes for JMS, see [Using IBM MQ classes for JMS in IMS](#).

Current releases of IMS support preloading IBM MQ modules from PDS-E format libraries in MPP, BMP, IFP, JMP and JBP regions only. Any other type of IMS region does not support preloading from PDS-E libraries. If preloading is required for any other type of region, then the IBM MQ modules that are provided must be copied to a PDS format library.

Defining IBM MQ queue managers to the IMS adapter

The names of the IBM MQ queue managers and their corresponding language interface tokens (LITs) must be defined in the queue manager definition table.

Use the supplied CSQQDEFX macro to create the CSQQDEFV load module. [Figure 115 on page 1044](#) shows the syntax of this assembler macro.

```
CSQQDEFX TYPE=ENTRY|DEFAULT,NAME=qmgr-name,LIT=token
or
CSQQDEFX TYPE=END
```

Figure 115. CSQQDEFX macro syntax

Parameters

TYPE=ENTRY|DEFAULT

Specify either TYPE=ENTRY or TYPE=DEFAULT as follows:

TYPE=ENTRY

Specifies that a table entry describing an IBM MQ queue manager available to an IMS application is to be generated. If this is the first entry, the table header is also generated, including a CSQQDEFV CSECT statement.

TYPE=DEFAULT

As for TYPE=ENTRY. The queue manager specified is the default queue manager to be used when MQCONN or MQCONNX specifies a name that is all blanks. There must be only one such entry in the table.

NAME= qmgr-name

Specifies the name of the queue manager, as specified with **MQCONN** or **MQCONNX**.

LIT= token

Specifies the name of the language interface token (LIT) that IMS uses to identify the queue manager.

An MQCONN or MQCONNX call associates the *name* input parameter and the *hconn* output parameter with the name label and, therefore, the LIT in the CSQQDEFV entry. Further IBM MQ calls passing the *hconn* parameter use the LIT from the CSQQDEFV entry identified in the MQCONN or MQCONNX call to direct calls to the IBM MQ queue manager defined in the IMS SSM PROCLIB member with that same LIT.

In summary, the **name** parameter on the MQCONN or MQCONNX call identifies a LIT in CSQQDEFV and the same LIT in the SSM member identifies an IBM MQ queue manager. (For information about the MQCONN call, see [MQCONN - Connect queue manager](#). For information about the MQCONNX call, see [MQCONNX - Connect queue manager \(extended\)](#).)

TYPE=END

Specifies that the table is complete. If this parameter is omitted, TYPE=ENTRY is assumed.

Using the CSQQDEFX macro

Figure 116 on page 1045 shows the general layout of a queue manager definition table.

```
CSQQDEFX NAME=subsystem1,LIT=token1
CSQQDEFX NAME=subsystem2,LIT=token2,TYPE=DEFAULT
CSQQDEFX NAME=subsystem3,LIT=token3
...
CSQQDEFX NAME=subsystemN,LIT=tokenN
CSQQDEFX TYPE=END
END
```

Figure 116. Layout of a queue manager definition table

Setting up the IMS trigger monitor

You can set up an IMS batch-oriented program to monitor an IBM MQ initiation queue.

Define the application to IMS using the model CSQQTAPL in the thlqual.SCSQPROC library (see [Example transaction definition for CSQQTRMN](#)).

Generate the PSB and ACB using the model CSQQTPSB in the thlqual.SCSQPROC library (see [Example PSB definition for CSQQTRMN](#)).

```
* This is the application definition *
* for the IMS Trigger Monitor BMP *

APPLCTN PSB=CSQQTRMN,
PGMTYPE=BATCH,
SCHDTYP=PARALLEL
```

Figure 117. Example transaction definition for CSQQTRMN

```

PCB TYPE=TP,           ALTPCB for transaction messages
MODIFY=YES,           To "triggered" IMS transaction
PCBNAME=CSQQTRMN
PCB TYPE=TP,           ALTPCB for diagnostic messages
MODIFY=YES,           To LTERM specified or "MASTER"
PCBNAME=CSQQTRMG,
EXPRESS=YES
PSBGEN LANG=ASSEM,
PSBNAME=CSQQTRMN,     Runs program CSQQTRMN
CMPAT=YES

```

Figure 118. Example PSB definition for CSQQTRMN

For further information about starting and stopping the IMS trigger monitor, see [Controlling the IMS trigger monitor](#).

Setting up the IMS bridge

The IBM MQ - IMS bridge is an optional component that enables IBM MQ to input and output to and from existing programs and transactions that are not IBM MQ-enabled.

This topic describes what you must do to customize the IBM MQ - IMS bridge.

Define the XCF and OTMA parameters for IBM MQ.

This step defines the XCF group and member names for your IBM MQ system, and other OTMA parameters. IBM MQ and IMS must belong to the same XCF group. Use the OTMACON keyword of the CSQ6SYSP macro to tailor these parameters in the system parameter load module.

See [Using CSQ6SYSP](#) for more information.

Define the XCF and OTMA parameters to IMS.

This step defines the XCF group and member names for the IMS system. IMS and IBM MQ must belong to the same XCF group.

Add the following parameters to your IMS parameter list, either in your JCL or in member DFSPBxxx in the IMS PROCLIB:

OTMA=Y

This starts OTMA automatically when IMS is started. (It is optional, if you specify OTMA=N you can also start OTMA by issuing the IMS command /START OTMA.)

GRNAME=

This parameter gives the XCF group name.

It is the same as the group name specified in the storage class definition (see the next step), and in the **Group** parameter of the OTMACON keyword of the CSQ6SYSP macro.

OTMANM=

This parameter gives the XCF member name of the IMS system.

This is the same as the member name specified in the storage class definition (see the next step).

Tell IBM MQ the XCF group and member name of the IMS system.

This is specified by the storage class of a queue. If you want to send messages across the IBM MQ - IMS bridge you must specify this when you define the storage class for the queue. In the storage class, you must define the XCF group and the member name of the target IMS system. To do this, either use the IBM MQ operations and control panels, or use the IBM MQ commands as described in [Introduction to Programmable Command Formats](#).

Set up the security that you require.

The /SECURE OTMA IMS command determines the level of security to be applied to **every** IBM MQ queue manager that connects to IMS through OTMA. See [Security considerations for using IBM MQ with IMS](#) for more information.

Adding an additional IMS connection to the same queue manager

To add an IMS connection to the same queue manager you must define a second storage class (STGCLASS) to point at the new IMS; see [DEFINE STGCLASS](#) for more information.

Important:

- One local queue cannot point to two storage classes.
- One storage class cannot point to two IMS bridges.
- IBM MQ and IMS must belong to the same XCF group. Use the OTMACON keyword of the CSQ6SYSP macro to tailor these parameters in the system parameter load module.

See [Using CSQ6SYSP](#) for more information.

Related concepts

[IBM MQ and IMS](#)

[“Setting up the IMS adapter” on page 1039](#)

To use IBM MQ within IMS requires the IBM MQ - IMS adapter (generally referred to as the IMS adapter).

[IMS and IMS bridge applications on IBM MQ for z/OS](#)

z/OS

Using IBM MQ with CICS

To use IBM MQ with CICS, you must configure the IBM MQ CICS adapter and, optionally, the IBM MQ CICS bridge components.

For more information about configuring the IBM MQ CICS adapter and the IBM MQ CICS bridge components, see the [Configuring connections to MQ](#) section of the CICS documentation.

Related concepts

[IBM MQ and CICS](#)

[“Using IBM MQ with IMS” on page 1039](#)

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

Related reference

[“Upgrading and applying service to Language Environment or z/OS Callable Services” on page 1047](#)

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

z/OS

Upgrading and applying service to Language Environment or z/OS Callable Services

The actions you must take vary according to whether you use CALLLIBS or LINK, and your version of SMP/E.

The following tables show you what you need to do to IBM MQ for z/OS if you upgrade your level of, or apply service to, the following products:

- Language Environment
- z/OS Callable Services (APPC and RRS for example)

Table 69. Service has been applied or the product has been upgraded to a new release

Product	Action if using CALLLIBS and SMP/E V3r2 or later Note: You do not need to run separate jobs for Language Environment and Callable services. One job will suffice.	Action if using LINK
Language Environment	<ol style="list-style-type: none"> 1. Set the Boundary on your SMP/E job to the Target zone. 2. On the SMP_CNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See <i>z/OS SMP/E Commands</i> for further information. 3. Run the SMP/E job. 	No action required provided that the SMP/E zones were set up for automatic relinking, and the CSQ8SLDQ job has been run.
Callable Services	<ol style="list-style-type: none"> 1. Set the Boundary on your SMP/E job to the Target zone. 2. On the SMP_CNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See <i>z/OS SMP/E Commands</i> for further information. 3. Run the SMP/E job. 	No action required provided that the SMP/E zones were set up for automatic relinking, and the CSQ8SLDQ job has been run.

Table 70. One of the products has been updated to a new release in a new SMP/E environment and libraries

Product	Action if using CALLLIBS and SMP/E V3r2 or later Note: You do not need to run three separate jobs for Language Environment and Callable services. One job will suffice for both products.	Action if using LINK
Language Environment	<ol style="list-style-type: none"> 1. Change the DDDEFs for SCEELKED and SCEESPC to point to the new library. 2. Set the Boundary on your SMP/E job to the Target zone. 3. On the SMP_CNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See <i>z/OS SMP/E Commands</i> for further information. 4. Run the SMP/E job. 	<ol style="list-style-type: none"> 1. Delete the XZMOD subentries for the following LMOD entries in the IBM MQ for z/OS target zone: CMQXDCST, CMQXRCTL, CMQXSUPR, CSQCBE00, CSQCBE30, CSQCBP00, CSQCBP10, CSQCBR00, CSQUCVX, CSQUDLQH, CSQVXPCB, CSQVXSPT, CSQXDCST, CSQXRCTL, CSQXSUPR, CSQXTDMI, CSQXTCP, CSQXTNSV, CSQ7DRPS, IMQB23IC, IMQB23IM, IMQB23IR, IMQS23IC, IMQS23IM, IMQS23IR 2. Set up the appropriate ZONEINDEXs between the IBM MQ zones and the Language Environment zones. 3. Tailor CSQ8SLDQ to refer to the new zone on the FROMZONE parameter of the LINK commands. CSQ8SLDQ can be found in the SCSQINST library. 4. Run CSQ8SLDQ.

Table 70. One of the products has been updated to a new release in a new SMP/E environment and libraries (continued)

Product	Action if using CALLLIBS and SMP/E V3r2 or later Note: You do not need to run three separate jobs for Language Environment and Callable services. One job will suffice for both products.	Action if using LINK
Callable services	<ol style="list-style-type: none"> 1. Change the DDDEF for CSSLIB to point to the new library 2. Set the Boundary on your SMP/E job to the Target zone. 3. On the SMP_CNTL card specify LINK LMODS CALLLIBS. You can also specify other parameters such as CHECK, RETRY(YES) and RC. See <i>z/OS SMP/E Commands</i> for further information. 4. Run the SMP/E job. 	<ol style="list-style-type: none"> 1. Delete the XZMOD subentries for the following LMOD entries in the IBM MQ for z/OS target zone: CMQXRCTL, CMQXSUPR, CSQBSRV, CSQILPLM, CSQXJST, CSQXRCTL, CSQXSUPR, CSQ3AMGP, CSQ3EPX, CSQ3REPL 2. Set up the appropriate ZONEINDEXs between the IBM MQ zones and the Callable Services zones. 3. Tailor CSQ8SLDQ to refer to the new zone on the FROMZONE parameter of the LINK commands. CSQ8SLDQ can be found in the SCSQINST library. 4. Run CSQ8SLDQ.

For an example of a job to relink modules when using CALLLIBS, see [“Running a LINK CALLLIBS job”](#) on page 1049.

▶ z/OS Running a LINK CALLLIBS job

An example job to relink modules when using CALLLIBS.

The following is an example of the job to relink modules when using CALLLIBs on a SMP/E V3r2 system. You must provide a JOBCARD and the data set name of SMP/E CSI that contains IBM MQ for z/OS.

```

//*****
//* RUN LINK CALLLIBS.
//*****
//CALLLIBS EXEC PGM=GIMSMP,REGION=4096K
//SMPCSI DD DSN=your.csi
//      DISP=SHR
//SYSPRINT DD SYSOUT=*
//SMPCNTL DD *
SET BDY(TZONE).
LINK LMODS CALLLIBS .
/*

```

Figure 119. Example SMP/E LINK CALLLIBS job

▶ z/OS Using OTMA exits in IMS

Use this topic if you want to use IMS Open Transaction Manager Access exits with IBM MQ for z/OS.

If you want to send output from an IMS transaction to IBM MQ, and that transaction did not originate in IBM MQ, you need to code one or more IMS OTMA exits.

Similarly if you want to send output to a non-OTMA destination, and the transaction did originate in IBM MQ, you also need to code one or more IMS OTMA exits.

The following exits are available in IMS to enable you to customize processing between IMS and IBM MQ:

- An OTMA pre-routing exit
- A destination resolution user (DRU) exit

OTMA exit names

You must name the pre-routing exit DFSYPRX0. You can name the DRU exit anything, as long as it does not conflict with a module name already in IMS.

Specifying the destination resolution user exit name

You can use the *Druexit* parameter of the OTMACON keyword of the CSQ6SYSP macro to specify the name of the OTMA DRU exit to be run by IMS.

To simplify object identification, consider adopting a naming convention of DRU0xxxx, where xxxx is the name of your IBM MQ queue manager.

If you do not specify the name of a DRU exit in the OTMACON parameter, the default is DFSYDRU0. See [DFSYDRU0](#) for more information.

Naming convention for IMS destination

You need a naming convention for the destination to which you send the output from your IMS program. This is the destination that is set in the CHNG call of your IMS application, or that is preset in the IMS PSB.

A sample scenario for an OTMA exit

Use the following topics for an example of a pre-routing exit and a destination routing exit for IMS:

- [“The pre-routing exit DFSYPRX0” on page 1050](#)
- [“The destination resolution user exit” on page 1051](#)

To simplify identification, make the OTMA destination name similar to the IBM MQ queue manager name, for example the IBM MQ queue manager name repeated. In this case, if the IBM MQ queue manager name is " **VCPE** ", the destination set by the CHNG call is " **VCPEVCPE** ".

Related concepts

[IBM MQ and IMS](#)

[“Using IBM MQ with IMS” on page 1039](#)

The IBM MQ -IMS adapter, and the IBM MQ - IMS bridge are the two components which allow IBM MQ to interact with IMS.

[IMS and IMS bridge applications on IBM MQ for z/OS](#)

The pre-routing exit DFSYPRX0

This topic contains a sample pre-routing exit for OTMA in IMS.

You must first code a pre-routing exit DFSYPRX0. See [OTMA Destination Resolution user exit \(DFSYPRX0 and other OTMAYPRX type exits\)](#) for parameters passed to this routine by IMS.

This exit tests whether the message is intended for a known OTMA destination (in our example VCPEVCPE). If it is, the exit must check whether the transaction sending the message originated in OTMA. If the message originated in OTMA, it will have an OTMA header, so you should exit from DFSYPRX0 with register 15 set to zero.

- If the transaction sending the message did not originate in OTMA, you must set the client name to be a valid OTMA client. This is the XCF member-name of the IBM MQ queue manager to which you want to send the message. You should set your client name (in the OTMACON parameter of the CSQ6SYSP

macro) is set to the queue manager name. This is the default. You should then exit from DFSYPRX0 setting register 15 to 4.

- If the transaction sending the message originated in OTMA, and the destination is non-OTMA, you should set register 15 to 8 and exit.
- In all other cases, you should set register 15 to zero.

If you set the OTMA client name to one that is not known to IMS, your application CHNG or ISRT call returns an A1 status code.

For an IMS system communicating with more than one IBM MQ queue manager, you should repeat the logic for each IBM MQ queue manager.

Sample assembler code is shown in [Figure 120 on page 1051](#):

```
TITLE 'DFSYPRX0: OTMA PRE-ROUTING USER EXIT'
DFSYPRX0 CSECT
DFSYPRX0 AMODE 31
DFSYPRX0 RMODE ANY
*
SAVE (14,12),,DFSYPRX0&SYSDATE&SYSTEMTIME
SPACE 2
LR R12,R15          MODULE ADDRESSABILITY
USING DFSYPRX0,R12
*
L R2,12(,R1)        R2 -> OTMA PREROUTE PARMS
*
LA R3,48(,R2)       R3 AT ORIGINAL OTMA CLIENT (IF ANY)
CLC 0(16,R3),=XL16'00' OTMA ORIG?
BNE OTMAIN          YES, GO TO THAT CODE
*
NOOTMAIN DS 0H      NOT OTMA INPUT
LA R5,8(,R2)        R5 IS AT THE DESTINATION NAME
CLC 0(8,R5),=C'VCPEVCPE' IS IT THE OTMA UNSOLICITED DEST?
BNE EXIT0           NO, NORMAL PROCESSING
*
L R4,80(,R2)        R4 AT ADDR OF OTMA CLIENT
MVC 0(16,R4),=CL16'VCPE' CLIENT OVERRIDE
B EXIT4             AND EXIT
*
OTMAIN DS 0H        OTMA INPUT
LA R5,8(,R2)        R5 IS AT THE DESTINATION NAME
CLC 0(8,R5),=C'VCPEVCPE' IS IT THE OTMA UNSOLICITED DEST?
BNE EXIT8           NO, NORMAL PROCESSING

*
EXIT0 DS 0H
LA R15,0            RC = 0
B BYEBYE
*
EXIT4 DS 0H
LA R15,4            RC = 4
B BYEBYE
*
EXIT8 DS 0H
LA R15,8            RC = 8
B BYEBYE
*
BYEBYE DS 0H
RETURN (14,12),,RC=(15) RETURN WITH RETURN CODE IN R15
SPACE 2
REQUATE
SPACE 2
END
```

Figure 120. OTMA pre-routing exit assembler sample

The destination resolution user exit

This topic contains a sample destination resolution user exit for IMS.

If you have set registers 15 to 4 in DFSYPRX0, or if the source of the transaction was OTMA **and** you set Register 15 to zero, your DRU exit is invoked. In this example, the DRU exit name is DRU0VCPE.

The DRU exit checks if the destination is VCPEVCPE. If it is, it sets the OTMA user data (in the OTMA prefix) as follows:

Offset

OTMA user data

(decimal)

0

OTMA user data length (in this example, 334)

2

MQMD

326

Reply to format

These offsets are where the IBM MQ - IMS bridge expects to find this information.

The DRU exit should be as simple as possible. Therefore, in this sample, all messages originating in IMS for a particular IBM MQ queue manager are put to the same IBM MQ queue.

If the message needs to be persistent, IMS must use a synchronized transaction pipe. To do this, the DRU exit must set the OUTPUT flag. See [Specifying synchronized tpipes for IBM MQ](#) for more information.

Write an IBM MQ application to process this queue, and use information from the MQMD structure, the MQIIH structure (if present), or the user data, to route each message to its destination.

A sample assembler DRU exit is shown in [Figure 121 on page 1053](#).

```

TITLE 'DRU0VCPE: OTMA DESTINATION RESOLUTION USER EXIT'
DRU0VCPE CSECT
DRU0VCPE AMODE 31
DRU0VCPE RMODE ANY
*
SAVE (14,12),,DRU0VCPE&SYSDATE&SYSTEME
SPACE 2
LR R12,R15          MODULE ADDRESSABILITY
USING DRU0VCPE,R12
*
L R2,12(,R1)        R2 -> OTMA DRU PARMS
*
L R5,88(,R2)        R5 ADDR OF OTMA USERDATA
LA R6,2(,R5)        R6 ADDR OF MQMD
USING MQMD,R6       AS A BASE
*
LA R4,MQMD_LENGTH+10 SET THE OTMA USERDATA LEN
STH R4,0(,R5)       = LL + MQMD + 8
*                   CLEAR REST OF USERDATA
MVI 0(R6),X'00'     ...NULL FIRST BYTE
MVC 1(255,R6),0(R6) ...AND PROPAGATE IT
MVC 256(MQMD_LENGTH-256+8,R6),255(R6) ...AND PROPAGATE IT
*
VCPE DS 0H
CLC 44(16,R2),=CL16'VCPE' IS DESTINATION VCPE?
BNE EXIT4           NO, THEN DEST IS NON-OTMA
MVC MQMD_REPLYTOQ,=CL48'IMS.BRIDGE.UNSOLICITED.QUEUE'
MVC MQMD_REPLYTOQMGR,=CL48'VCPE' SET QNAME AND QMGRNAME
MVC MQMD_FORMAT,MQFMT_IMS SET MQMD FORMAT NAME
MVC MQMD_LENGTH(8,R6),MQFMT_IMS_VAR_STRING
*                   SET REPLYTO FORMAT NAME
B EXIT0
*
EXIT0 DS 0H
LA R15,0            SET RC TO OTMA PROCESS
B BYEBYE           AND EXIT
*
EXIT4 DS 0H
LA R15,4            SET RC TO NON-OTMA
B BYEBYE           AND EXIT
*
BYEBYE DS 0H
RETURN (14,12),,RC=(15) RETURN CODE IN R15
SPACE 2
REQUATE
SPACE 2
CMQA EQUONLY=NO
CMQMDA DSECT=YES
SPACE 2
END

```

Figure 121. Sample assembler DRU exit

z/OS

Using IBM z/OSMF to automate IBM MQ

The IBM z/OS Management Facility (z/OSMF) provides system management functions in a task-oriented, web browser-based user interface with integrated user assistance, so that you can more easily manage the day-to-day operations and administration of your mainframe z/OS systems.

By streamlining some traditional tasks and automating others, z/OSMF can help to simplify some areas of z/OS system management.

Resources can be provisioned or de-provisioned, at a click of a button, from a user provided portal. z/OSMF provides REST APIs to help with this task.

The sample marketplace portal supplied with z/OSMF can also be used to provision and de-provision resources. Alternatively, more experienced users can use the z/OSMF Web User Interface (WUI).

This section assumes that you understand z/OSMF, but if you are unfamiliar with z/OSMF you should read [Getting started with z/OSMF](#). Alternatively, you can access this section from the z/OSMF WUI online help.

You should familiarize yourself with z/OS Cloud configuration, that is:

- Cloud Provisioning - [Resource management services](#)
- Workload Management - see [IBM z/OS Management Facility Programming Guide](#) for more information.
- Getting started - see [Getting Started Tutorial - Cloud](#)

z/OSMF 2.2 introduces role based activities and tasks, so it is important that you understand concepts like:

- domains
- administrators
- approvers
- tenants
- templates
- instances
- workflows

and so on.

Sample IBM MQ z/OSMF workflows and associated files are provided, and can be installed as part of the IBM MQ for z/OS UNIX System Services Components feature. The installation process for this feature, and the directory and file structure, are described in the IBM MQ for z/OS Program Directory. Pour les liens de téléchargement des répertoires de programme, voir [IBM MQ for z/OS Program Directory PDF files](#).

The sample workflows are written in XML and demonstrate how to automate the provisioning (creation) or de-provisioning (destruction) of IBM MQ queue managers, channel initiators, and local queues, and how to perform actions against the provisioned IBM MQ resources. Steps within the workflows submit jobs (JCL), run REXX execs, process Shell scripts, or issue REST API calls.

The samples are designed to illustrate the types of function that can be achieved using z/OSMF. It is anticipated that z/OSMF workflows will generally be used to provision resources and actions like put or get message will, in essence, be performed using IBM MQ applications.

You can run the sample workflows as supplied, provided the workflow variable properties have been set (as discussed in the following sections), or you can customize them as required. You might prefer to write your own workflows to perform additional function. Before running the sample workflows see:

- [“Prerequisites for z/OSMF” on page 1054](#)
- [“Security settings ” on page 1056](#)
- [“Limitations ” on page 1058](#)

Sample workflow applications are provided to:

- [“Automate the provisioning or de-provisioning of IBM MQ queue managers and perform actions against the provisioned queue managers” on page 1059](#)
- [“Automate the provisioning or de-provisioning of IBM MQ local queues and perform actions against the provisioned queues” on page 1060.](#)

Related concepts

[“Setting up IBM MQ for z/OS” on page 935](#)

Use this topic as a step by step guide for customizing your IBM MQ for z/OS system .

Prerequisites for z/OSMF

The prerequisites you require to run IBM z/OS Management Facility (z/OSMF) with IBM MQ

The workflows shipped in IBM MQ for z/OS 9.1.0 exploit new function in z/OSMF, which is provided through APARs on both z/OS 2.1 and 2.2. More details are provided in the following text.

1. You have installed and configured IBM z/OS Management Facility 2.2 correctly. If you are running with security enabled, ensure that all security settings as documented by z/OSMF have been configured.

2. You have installed the following APARs for:

z/OS 2.1

- PI71068
- PI71079
- PI71082
- PI71084
- OA50130

z/OS 2.2

- PI70526
- PI70521
- PI70527
- PI67839
- PI70767
- PI46315
- OA49081
- OA49802
- OA50130

3. The z/OSMF angel (if required) and server processes have been configured.
4. The z/OS Cloud environment has been configured (as briefly discussed above and documented by z/OSMF)
5. IBM MQ for z/OS 9.0.1 has been installed and the product load libraries are available.
6. The following IBM MQ queue manager customization tasks have been performed:

Task	Description
1	Identify the z/OS system parameters
2	APF authorize the IBM MQ load libraries
3	Update the z/OS link list and LPA
4	Update the z/OS program properties table

7. The sample workflows and associated files are installed in a suitable z/OS UNIX System Services (z/OS UNIX) directory.
8. The /tmp z/OS UNIX directory is available, because the provision.xml workflow might create a temporary file in this directory. If a file is created, the workflow, in general, deletes the file after use.
9. The deprovision.xml file has steps in it that invoke the CSQ4ZWS1.rexx and CSQ4ZWS2.rexx REXX execs. These execs wait for the queue manager and channel initiator subsystems to stop; the execs invoke the z/OS UNIX **SLEEP** command as a system call.

Depending on your z/OS UNIX configuration, you might find that the **SLEEP** command does not work as coded. If, during processing you encounter an error which indicates that the **SLEEP** command cannot be found, you can try replacing the following lines in execs CSQ4ZWS1.rexx and CSQ4ZWS2.rexx:

```
CALL SYSCALLS('ON')           /* Enable z/OS UNIX calls */
ADDRESS SYSCALL
"SLEEP" 10                    /* Sleep for 10 seconds */
CALL SYSCALLS 'OFF'          /* Disable z/OS UNIX calls */
```

with

Then, issue the Open MVS (OMVS) **env** command to check your PATH environment variable setting. Ensure that the directory which contains the **sleep** command is defined to the PATH. Note that the **sleep** command is typically found in the /bin directory.

10. Ensure that z/OSMF has been started.

Both the angel and server z/OSMF processes must be started and the z/OSMF Web User Interface (WUI) be up and running. For further details, see [Liberty profile: Process types on z/OS](#).

Even if you intend to drive the workflows using the REST API, the z/OSMF WUI needs to be started. The z/OSMF WUI can be useful for monitoring the creation and execution of workflows.

Related concepts


“Using IBM z/OSMF to automate IBM MQ ” on page 1053

The IBM z/OS Management Facility (z/OSMF) provides system management functions in a task-oriented, web browser-based user interface with integrated user assistance, so that you can more easily manage the day-to-day operations and administration of your mainframe z/OS systems.

Security settings

The security settings required to run z/OSMF.

The following User ID variable properties are defined in the properties file. For more details, see “[Running the workflows](#)” on page 1063.

User ID property	Description
CSQ_USERID	User ID used to run the workflow steps. Note, however, that selected steps (which generally require an elevated level of authority) will be run with different user IDs based on the setting of the CSQ_ADMIN_* user IDs listed in the following text. The user ID in use is identified by the runAsUser property on the respective step in the workflows.
CSQ_ADMIN_APF_USERID	User ID to use when APF authorizing the load library that contains the queue manager system parameter module.
CSQ_APF_APPROVAL_ID	The approval ID used to permit users to run the data set APF authorization step as user CSQ_ADMIN_APF_USERID.
CSQ_ADMIN_CONSOLE_USERID	User ID used when running steps under the run that issue z/OS console commands.  Attention: This user ID needs to be permitted UPDATE access to the started task profile (MVS.START.STC.*) in the OPERCMDS class. See Controlling the use of operator commands in the z/OS documentation for more information.
CSQ_CONSOLE_APPROVAL_ID	The approval ID used to permit users to run steps that issue z/OS console commands under the run as user CSQ_ADMIN_CONSOLE_USERID.
CSQ_ADMIN_SAF_USERID	User ID to use when issuing SAF commands.
CSQ_SAF_APPROVAL_ID	The approval ID used to permit users to run the SAF command steps under the run as user CSQ_ADMIN_SAF_USERID.
CSQ_ADMIN_SSI_USERID	User ID to use when issuing the SETSSI command to identify the subsystem being provisioned to z/OS.
CSQ_SSI_APPROVAL_ID	The approval ID used to permit users to run the SETSSI command step under the run as user CSQ_ADMIN_SSI_USERID.

Note: The User ID being used to run the provision and de-provision workflows needs to have sufficient authority as listed below:

1. The Queue Manager provision and de-provision workflows use the SETPROG command to APF authorize data sets. Either the user ID is set in property CSQ_ADMIN_APF_USERID, or the user ID being used to run the workflows needs to be permitted to issue this command. You can achieve this by issuing the following command:

```
PERMIT MVS.SETPROG CLASS(OPERCMD5) ID(value of CSQ_ADMIN_APF_USERID) ACCESS(UPDATE)
```

Note: The SETPROG command might not persist across an IPL of a z/OS system so, it might be necessary to manually issue the following SETPROG command following an IPL:

```
SETPROG APF,ADD,DSN=value of CSQ_AUTH_LIB_HLQ.value of CSQ_SSID.APF.LOAD,SMS
```

For more details about the SETPROG command, see [Using RACF to control APF lists](#).

In addition, you might have enabled FACILITY class to control which libraries can be APF authorized, so you might need to issue the command:

```
PERMIT CSVAPF.libname CLASS(FACILITY) ID(value of CSQ_ADMIN_APF_USERID)  
ACCESS(UPDATE)
```

2. A step in the Queue Manager provision workflow issues the SETSSI command to identify the IBM MQ subsystem to z/OS. The User ID set in property CSQ_ADMIN_SSI_USERID needs to be permitted to use this command. You can achieve this by issuing the following command:

```
PERMIT MVS.SETSSI.ADD CLASS(OPERCMD5) ID(value of CSQ_ADMIN_SSI_USERID)  
ACCESS(CONTROL)
```

Note: Subsystems that have been identified to z/OS through the SETSSI command do not persist across an IPL of a z/OS system. So, it might be necessary to manually issue the following SETSSI command following an IPL:

```
SETSSI ADD,S=value of CSQ_SSID,I=CSQ3INI,  
P=CSQ3EPX,value of CSQ_CMD_PFX,S
```

For more details about the SETSSI command, see: [SETSSI command](#).

3. The workflows issue queue manager commands, so if you are planning to enable security, the user ID set in property CSQ_ADMIN_RACF_USERID (or the user ID being used to run the workflows) needs to be granted CLAUTH (client authentication) authority to the MQADMIN or the MXADMIN class (depending on which class is being used). This is to allow this user ID to define security profiles to these classes. You can achieve this by issuing the following command:

```
ALTUSR value of CSQ_ADMIN_RACF_USERID CLAUTH(MQADMIN)
```

For more details about **CLAUTH** see [The CLAUTH \(class authority\) attribute](#).

4. The deprovision.xml workflow issues z/OS commands, for example, DISPLAY ACTIVE jobs, CANCEL or FORCE subsystems, so the user ID set in property CSQ_ADMIN_CONSOLE_USERID (or the user ID being used to run the workflows) needs to have suitable authority to issue such commands.
5. Users requesting a queue manager instance, using the templates table of the Software Services task, must have permission to access z/OSMF and the Configuration Assistant, as defined by z/OSMF.
6. The user ID of the consumer provisioning a queue manager requires authority to add and delete members from the PROCLIB data set defined with variable CSQ_PROC_LIB.
7. A queue manager must be provisioned ahead of provisioning queues.

8. To use the `queueLoad.xml` and `queueOffload.xml` workflows, the data sets used need to be defined ahead of time. Also, the user ID used to run these workflows needs to be granted UPDATE authority to the data sets.
9. A step in the queue manager `provision.xml` workflow currently disables subsystem security. You can modify `Job csq4znse.jcl` to enable subsystem security by adding the appropriate security commands for protecting IBM MQ resources. However, note that if you do add additional commands, you also need to add commands to delete security permissions in `csq4dse.jcl`, which is submitted by the `deprovision.xml` workflow.

Note: This step issues RACF security commands. If you are using an alternate security product, you need to modify this step to issue the appropriate commands for your security product.

Network Requirements

When adding a queue manager template, and resources for the template, you need to click **Create network resource pool**. This creates a resource pool with network resources for this template.

Using the Configuration Assistant, your network administrator needs to complete this network resource pool definition by defining a limit for the number of ports that are to be allocated for this template.

For each template instance, the `provision.xml` workflow allocates a port in the range, and starts a listener to listen on that port.

Classifying with IBM Workload Manager

If you want to classify the queue manager and channel initiator address spaces with WLM, you need to specify this when adding a template for provisioning a queue manager.

Whether to classify or not, is controlled by flags **CSQ_DEFINE_MSTR_WLM_RULE** and **CSQ_DEFINE_CHIN_WLM_RULE**, which are set in file `workflow_variables.properties`.

For more information about classifying with WLM, refer to the *z/OSMF Configuration Guide*.

Related concepts

[“Prerequisites for z/OSMF” on page 1054](#)

The prerequisites you require to run IBM z/OS Management Facility (z/OSMF) with IBM MQ

Limitations

Limitations when using z/OSMF with IBM MQ.

1. The `provision.xml` workflow currently automates the following highlighted queue manager customization tasks:

Task	Description
1	Identify the z/OS system parameters
2	APF authorize the IBM MQ load libraries (provision.xml does APF authorize some libraries)
3	Update the z/OS link list and LPA
4	Update the z/OS program properties table
5	Define the IBM MQ subsystem to z/OS
6	Create procedures for the IBM MQ queue manager
7	Create procedures for the channel initiator
8	Define the IBM MQ subsystem to a z/OS WLM service class
9	Select and set up your coupling facility offload storage environment

Task	Description
10	Set up the coupling facility
11	Implement your ESM security controls
12	Update SYS1.PARMLIB members
13	Customize the initialization input data sets
14	Create the bootstrap and log data sets
15	Define your page sets
16	Add the IBM MQ entries to the Db2 data-sharing group
17	Tailor your system parameter modules (some)
18	Tailor the channel initiator parameters (some)
19	Set up Batch, TSO, and RRS adapters
20	Set up the operations and control panels
21	Include the IBM MQ dump formatting member
22	Suppress information messages
23	Update your system DIAG member for Advanced Message Security
24	Create procedures for Advanced Message Security
25	Set up the started task user Advanced Message Security
26	Grant RACDCERT permissions to the security administrator for Advanced Message Security
27	Grant users resource permissions for Advanced Message Security

2. Customization tasks that are not highlighted in bold text need to be performed manually, if required.
3. The sample INP1 and INP2 members are currently used as is. If required, additional properties can be defined to control the resources defined by these members.
4. Comments pertaining to specific properties listed in the properties file indicate any limitations of using those properties. For more details, see [“Running the workflows” on page 1063](#).

Related concepts

“Security settings ” on page 1056

The security settings required to run z/OSMF.

Automate the provisioning of IBM MQ objects

Samples are supplied to automate the provisioning of queue managers and local queues.

Automate the provisioning or de-provisioning of IBM MQ queue managers and perform actions against the provisioned queue managers

The following queue manager specific sample z/OSMF workflows are provided:

Workflow name	Description
provision.xml	Provision an IBM MQ for z/OS queue manager This sample workflow: <ul style="list-style-type: none"> • Provisions the required system resources for a queue manager.

Workflow name	Description
	<ul style="list-style-type: none"> Provisions the required system resources for a channel initiator. Starts the queue manager (which also starts the channel initiator and TCP/IP listener) Runs the sample queue manager installation verification program. <p>An environment property can be set to control the provisioning of queue managers with different characteristics. For more information, see “Running the workflows” on page 1063.</p> <p>Note: A manifest file (<code>provision.mf</code>) is provided to assist with adding a template for this workflow. This file contains a reference to the qaas_readme.pdf file which contains additional information. You can access the file through a link, once the template has been added.</p>
deprovision.xml	<p>De-provision an IBM MQ for z/OS queue manager</p> <p>This sample workflow:</p> <ul style="list-style-type: none"> Stops the channel initiator (which also stops the TCP/IP listener) and the queue manager. Waits for the subsystems to stop De-provisions all channel initiator and queue manager system resources.
startQMgr.xml	<p>Start an IBM MQ for z/OS queue manager</p> <p>This sample workflow starts the queue manager (which also starts the channel initiator and TCP/IP listener).</p>
stopQMgr.xml	<p>Stop an IBM MQ for z/OS queue manager</p> <p>This sample workflow stops the channel initiator (which also stops the TCP/IP listener) and the queue manager.</p>

Each workflow performs one or more steps. Comments in the workflows explain the function performed by each step. Some of the steps just request data input, while some steps submit JCL, invoke REXX execs, Shell scripts, or issue REST API calls to accomplish the stated function.

Refer to each step for the exact name of the JCL or REXX exec files. The workflows and associated JCL or REXX exec files reference variables that are declared in one or more variable XML files. For more details, see [“Workflow variable declaration files”](#) on page 1062.

deprovision, **startQMgr**, and **stopQMgr** can be performed as actions against a provisioned IBM MQ for z/OS queue manager.

Automate the provisioning or de-provisioning of IBM MQ local queues and perform actions against the provisioned queues

The following queue specific sample z/OSMF workflows are provided:

Workflow name	Description
defineQueue.xml	<p>Define a local queue</p> <p>This sample workflow demonstrates how z/OSMF workflows can be used to define small, medium, or large sized queues based on property settings.</p> <p>Note: A manifest file (<code>provision.mf</code>) is provided to assist with adding a template for this workflow. This file contains a reference to the</p>

Workflow name	Description
	qaas_readme.pdf file which contains additional information. You can access the file through a link, once the template has been added.
displayQueue.xml	<p>Display selected attributes of a local queue</p> <p>This sample workflow displays selected attributes of a local queue. The attributes are returned in a z/OSMF variable (refer to the steps in the workflow for the name of the variable) and subsequently displayed. If required, the contents of the variable can be accessed using a REST API.</p> <p>For more details, refer to Cloud provisioning REST APIs, and also see z/OSMF workflow services.</p>
deleteQueue.xml	<p>Delete a local queue</p> <p>This sample workflow deletes a local queue on a specified queue manager.</p>
putQueue.xml	<p>Put one or more messages to a local queue.</p> <p>This sample workflow puts one or more messages to a local queue. The message text can be specified but if more than one message is put to a local queue at the same time, the same message text is used.</p>
getQueue.xml	<p>Get one or more messages from a local queue.</p> <p>This sample workflow gets one or more messages from a local queue. The messages are returned in a z/OSMF variable (refer to the steps in the workflow for the name of the variable) and subsequently displayed. If required, you can access the contents of the variable using a REST API.</p> <p>For more details, refer to Cloud provisioning REST APIs, and also see z/OSMF workflow services.</p>
loadQueue.xml	<p>Load messages from a data set to a local queue.</p> <p>This sample workflow loads messages from a data set on to a local queue. The default name of the data set is specified by setting a property. For more details, see “Running the workflows” on page 1063.</p>
offloadQueue.xml	<p>Offload messages from a local queue to a data set.</p> <p>This sample workflow off-loads messages from a local queue to a data set. The default name of the data set is specified by setting a property. For more details, see “Running the workflows” on page 1063.</p>
clearQueue.xml	<p>Clear messages on a local queue.</p> <p>This sample workflow clears (deletes) all messages on a local queue.</p>

Notes:

1. The **Put Queue** action allows you to enter some message data and put one or more messages onto a queue. If more than one message is to be placed onto a queue during a given request, the same message data is used.
2. The loadQueue.xml and offloadQueue.xml workflows invoke the executable module, CSQUDMSG in the SCSQLOAD library, with an alias of QLOAD. This is equivalent to the **dmpmqmsg** utility available with IBM MQ for Multiplatforms. Therefore messages loaded from a data set onto a queue, or from a queue onto a data set, are expected to be in the **dmpmqmsg** format.

Sample JCL is also provided as member CSQ4QLOD in SCSQPROC.

The easiest way to try out the loadQueue and offloadQueue actions is to do the following:

- a. Issue **putQueue** a few times to put some messages on to a queue.
- b. Use **offloadQueue** to offload the messages from the queue on to a data set.
- c. If required, issue **clearQueue** to remove all messages from the queue.
- d. Use **loadQueue** to load the messages from a data set onto the same or a different queue.

If you are interested in the **dmpmqmsg** format, you can browse the contents of the data set, once you have issued an Offload request.

3. You can perform **displayQueue**, **deleteQueue**, **putQueue**, **getQueue**, **loadQueue**, **offloadQueue**, and **clearQueue** as actions against a provisioned IBM MQ for z/OS local queue. For further details about actions and action files, refer to the *z/OSMF Programming Guide*.
4. All action related workflows are deleted by default. The reason for this is to minimize the need for users to cleanup workflows.

The problem with this however is that where an action results in some output. For example, the **displayQueue** and **getQueue** actions both produce output.

The output cannot be seen since the related workflow is deleted as soon as the action has been performed. So, if you drive the workflow actions from the z/OS WUI, you need to set the **cleanAfterComplete** flag to *false* on the **<workflow>** tag for each action whose output you want to see.

For example, to see the output of **displayQueue**, set the flag as follows:

```
<action name="displayQueue">
  <workflow cleanAfterComplete="false">
    ...
  </workflow>
</action>
```

However, this means that you then have to manually clean up action related workflows.

Each sample z/OSMF workflow performs one or more steps. Comments in the workflows explain the function performed by each step. Some of the steps just request data input while some steps submit JCL and others invoke REXX execs to accomplish the stated function.

Refer to each step for the exact name of the JCL or REXX exec files. The workflows and associated JCL or REXX exec files reference variables that are declared in one or more [“Workflow variable declaration files”](#) on page 1062.

Related concepts

[“Limitations ” on page 1058](#)

Limitations when using z/OSMF with IBM MQ.

Running workflows

A description of the files referenced by the sample The z/OSMF workflows, and how you run a workflow.

Workflow variable declaration files

The following files declare variables that are referenced by the sample z/OSMF workflows and associated JCL or REXX exec files:

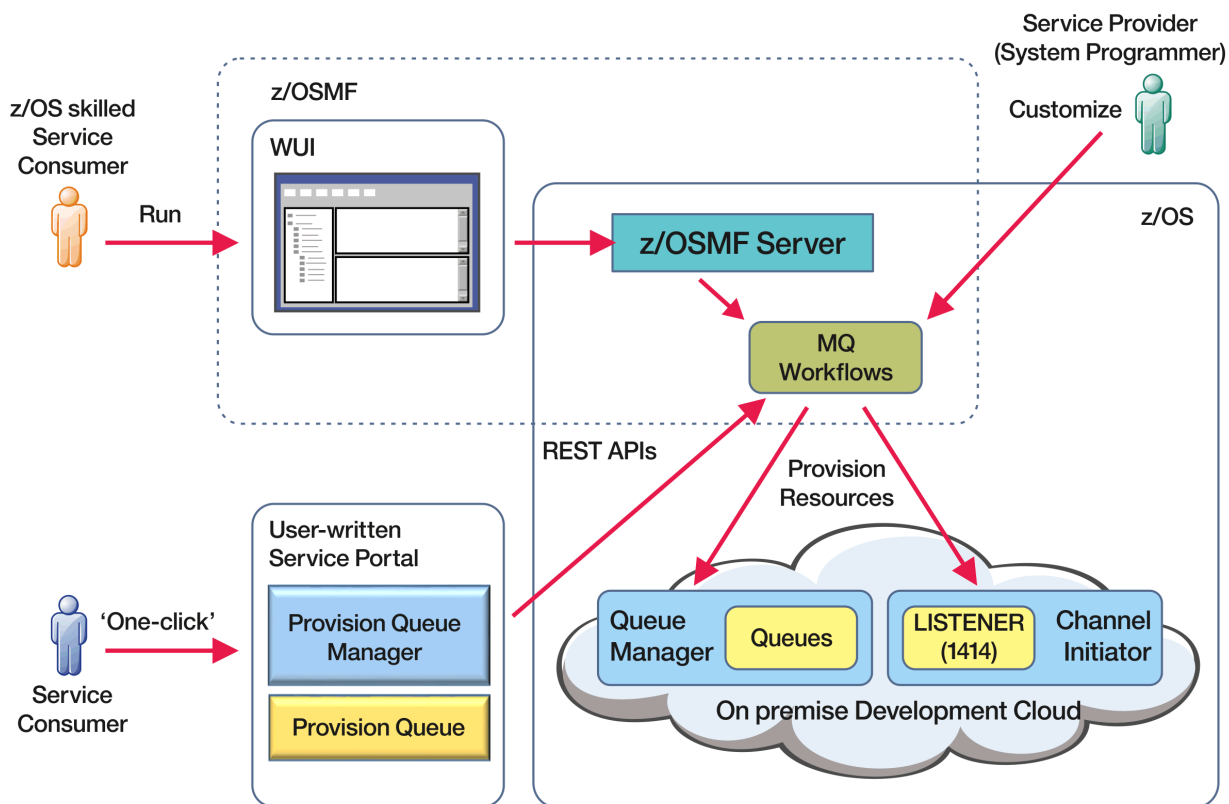
Workflow variable declaration file name	Description
common_variables.xml	Variables common to both the queue manager (plus channel initiator) and queue workflows.
qmgr_variables.xml	Variables specific to the queue manager (plus channel initiator) workflows.

Workflow variable declaration file name	Description
queue_variables.xml	Variables specific to the queue workflows.
tcpip_variables.xml	Variables specific to the queue manager (plus channel initiator) workflows, and used for identifying TCP/IP resources.

Note: The default visibility of variables is *private*. To allow variables to be queried using the z/OSMF REST API, selected variables have been marked as *public*. However, you can change the visibility of a given variable if required.

Running the workflows

Figure 122. 'One-click' provisioning of IBM MQ for z/OS resources



Before the workflows can be run, some properties need to be set in the following file:

Workflow variable properties file name	Description
workflow_variables.properties	<p>Initial properties for the workflow variables. Comments in the file indicate the purpose of each property.</p> <ul style="list-style-type: none"> Properties within meta-brackets (< >) need to be set to user specific values. An environment property can be set to provision queue managers for development (DEV), or test (TEST), or quality assurance (QA), or production (PROD) environments.

Workflow variable properties file name	Description
	<p>Additional property settings control the characteristics of the queue manager to be provisioned for each environment. For example you can vary the number of active logs, or the number of page sets, for each environment type.</p> <ul style="list-style-type: none"> • Other properties are set to IBM MQ default values but can be modified to meet local conventions if required.

In general, once the properties have been set, the workflows can be run as is. However, if required, you can customize a workflow to modify or remove existing steps, or to add new steps.

Workflows can be run:

- From the z/OSMF WUI.

From Cloud Provisioning -> Software Services in the WUI, workflows can be run in automatic or manual mode. The manual mode is useful when testing, and in both modes the progress of each step in the workflow can be monitored.

For more details, see [Cloud provisioning services](#) and [Create a workflow](#).

- Using the z/OSMF REST Workflow Services.

The REST Workflow Services can be used to run workflows through a REST API. This mode is useful for creating one-click operations from a user-written portal.

For more details, refer to [Cloud provisioning REST APIs](#), and also see [z/OSMF workflow services](#).

- Using the sample marketplace portal provided with z/OSMF.

Related concepts

“Automate the provisioning of IBM MQ objects” on page 1059

Samples are supplied to automate the provisioning of queue managers and local queues.

z/OS MQ Adv. VUE Activation des agents MFT pour la connexion à des gestionnaires de files d'attente z/OS éloignées

Sous réserve d'autorisation, un agent MFT sous z/OS peut utiliser une connexion client pour se connecter à un gestionnaire de files d'attente z/OS. Cela peut conduire à des topologies IBM MQ plus simples.

Si un agent MFT sous z/OS est associé à l'identificateur de produit (PID) de IBM MQ Advanced for z/OS VUE ou IBM MQ Advanced for z/OS, l'agent peut utiliser une connexion client pour se connecter à un gestionnaire de files d'attente sous z/OS.

Pour plus d'informations sur les différents PID, voir [Identificateurs de produit IBM MQ et informations d'exportation](#). Pour plus d'informations sur la définition du PID associé à une installation MFT, voir [fteSetProductId](#).

Le PID sous lequel l'agent s'exécute est affiché dans le journal au démarrage de l'agent.

Un agent MFT sous z/OS, s'exécutant sous un autre PID, ne peut se connecter qu'à un gestionnaire de files d'attente local à l'aide d'une connexion en mode liaisons.

Si un agent tente de se connecter à un gestionnaire de files d'attente qui ne s'exécute pas sous z/OS, le message BFGQM1044E est émis et le démarrage de l'agent est arrêté.

Tâches associées

[Démarrage d'un agent MFT sous z/OS](#)

[Arrêt d'un agent MFT sous z/OS](#)

Référence associée

[Le fichier MFT agent.properties](#)

Configuration de IBM MQ Internet Pass-Thru

Cette section décrit les différentes fonctions prises en charge par IBM MQ Internet Pass-Thru (MQIPT) et explique comment les configurer.

Configurez MQIPT en modifiant le fichier de configuration `mqipt.conf`. La structure du fichier de configuration MQIPT et les propriétés pouvant être spécifiées sont décrites dans la rubrique [IBM MQ Internet Pass-Thru -Référence pour la configuration](#).

Remarque : Vous devez définir des droits d'accès aux fichiers sécurisés sur le répertoire dans lequel se trouve le fichier `mqipt.conf` pour empêcher les utilisateurs non autorisés de voir les mots de passe stockés ou de modifier la configuration. Protégez tous les mots de passe spécifiés dans le fichier de configuration en suivant la procédure décrite dans [«Chiffrement des mots de passe stockés dans MQIPT»](#), à la page 1105.

Les modifications apportées au fichier de configuration prennent effet lorsque MQIPT est démarré ou actualisé. L'actualisation d'une instance active de MQIPT permet d'appliquer les modifications de configuration sans redémarrer MQIPT. Lorsque MQIPT est actualisé, le fichier de configuration `mqipt.conf` est relu et MQIPT effectue les actions suivantes:



- Toutes les routes actives marquées comme étant inactives ou qui ne sont plus spécifiées dans le fichier de configuration sont fermées et n'acceptent plus les connexions entrantes.
- Toutes les routes qui sont marquées comme actives dans le fichier de configuration et qui ne sont pas en cours d'exécution sont démarrées.
- Toutes les modifications apportées aux paramètres de configuration des routes actives sont appliquées. Dans la mesure du possible, ces modifications prennent effet sans interruption des connexions actives. Pour certaines modifications de paramètre, telles qu'une modification de la destination de la route, toutes les connexions sont fermées avant que la modification ne soit appliquée et que la route ne soit redémarrée.

Pour actualiser MQIPT, utilisez la commande `mqiptAdmin`. Pour plus d'informations sur l'administration de MQIPT à l'aide de la commande `mqiptAdmin`, voir [Administration MQIPT à l'aide de la ligne de commande](#).

Prise en charge HTTP dans MQIPT

MQIPT prend en charge la tunnellation HTTP. MQIPT peut être configuré de sorte que les paquets de données qu'il réachemine soient codés en tant que requêtes HTTP.

Les canaux IBM MQ n'acceptent pas les demandes HTTP. Par conséquent, une seconde MQIPT est nécessaire pour recevoir les demandes HTTP et les reconvertir en paquets de protocole IBM MQ. Le second MQIPT supprime l'en-tête HTTP pour reconvertir le paquet entrant en un paquet de protocole IBM MQ standard, avant de le transmettre au gestionnaire de files d'attente de destination.

Remarque :   Depuis IBM MQ 9.4.0, les routes MQIPT n'acceptent pas les connexions HTTP par défaut. Les routes doivent être configurées pour accepter les connexions HTTP à l'aide de la propriété **AllowedProtocols**.

Lorsque HTTP est utilisé entre deux instances de MQIPT, la connexion TCP/IP sur lequel le flux des demandes et des réponses HTTP est persistante et reste ouverte pendant toute la durée de vie du canal de message. MQIPT ne ferme pas la connexion TCP/IP entre les paires de demandes/réponses.

Si deux instances de MQIPT communiquent via HTTP, une demande HTTP peut alors rester en attente pendant une période assez longue. On peut prendre l'exemple d'un canal demandeur/serveur, lorsque le côté serveur attend l'arrivée de nouveaux messages dans sa file d'attente de transmission. Le protocole de canal IBM MQ fournit un dispositif de signal de présence, ou pulsation de canal, qui exige que le destinataire en attente envoie régulièrement des messages de signaux de présence à son partenaire. L'intervalle de la pulsation de canal par défaut est de 5 minutes. MQIPT utilise cette pulsation comme réponse HTTP. Ne désactivez pas la pulsation de canal, et ne la définissez pas sur une valeur trop élevée pour éviter de provoquer des problèmes d'expiration de délais d'attente dans certains pare-feux.

MQIPT accepte le trafic HTTP au format tronqué, généré par un proxy ou un serveur HTTP.

Pour un exemple d'utilisation de HTTP dans MQIPT, voir [Configuration de la tunnellation HTTP](#).

Proxys HTTP

Un proxy HTTP peut être placé entre les deux instances de MQIPT. Le proxy HTTP doit répondre aux exigences suivantes:

- Le proxy doit prendre en charge le protocole HTTP 1.1 .
- Les en-têtes HTTP **Connection** ou **Proxy-Connection** définis par MQIPT doivent être traités par le proxy. Cela permet de conserver ouvertes les connexions entre les deux instances de MQIPT pendant la durée de vie du canal de transmission de messages.
- Un mappage un à un des connexions persistantes doit être géré sur le proxy. Cela permet de s'assurer que les connexions TCP/IP entre le proxy et la destination MQIPT ne sont pas utilisées pour transmettre des données pour plusieurs canaux de transmission de messages.

Vous pouvez définir des propriétés pour configurer la façon dont les connexions persistantes sont gérées sur certains proxys HTTP. Par exemple, vous pouvez définir le nombre maximal de demandes pouvant être effectuées sur une connexion permanente. Les propriétés suivantes doivent être définies:

- Les connexions persistantes doivent être activées.
- La réutilisation des connexions TCP/IP du proxy vers MQIPT par plusieurs sessions HTTP doit être désactivée afin de conserver un mappage un à un des connexions persistantes sur le proxy.
- Le délai d'attente sur les demandes de proxy doit être défini sur une valeur élevée. Par exemple, 12 heures.
- Le nombre maximal de demandes pouvant être effectuées sur une connexion permanente doit être défini sur une valeur élevée. Par exemple, 5000.

MQIPT utilise des demandes HTTP POST pour envoyer des données entre les deux instances de MQIPT. Si la configuration MQIPT spécifie le nom d'hôte du proxy à l'aide de la propriété **HTTProxy**, MQIPT se connecte au proxy et utilise la méthode HTTP CONNECT pour demander au proxy d'établir un tunnel vers la destination MQIPT. Cela permet aux connexions HTTPS de passer par le proxy sans mettre fin à la session TLS dans le proxy.

Si un équilibreur de charge est placé entre les instances MQIPT, il doit être configuré pour utiliser la valeur du cookie HTTP *MQIPTSessionId* afin de s'assurer que toutes les demandes de chaque session sont transmises à la même destination.

HTTPS dans MQIPT

HTTPS peut être utilisé sur une connexion HTTP en activant les propriétés de route **HTTPS** et **SSLClient** sur le serveur MQIPT qui émet la connexion client.

MQIPT doit avoir accès au certificat de l'autorité de certification sécurisée qui va être utilisé pour authentifier le serveur/proxy HTTP cible. La propriété **SSLClientCAKeyring** peut être utilisée pour définir le fichier de clés contenant le certificat de l'autorité de certification digne de confiance.

Une configuration courante de HTTPS consiste à utiliser un proxy HTTP local pour créer une tunnellation via un pare-feu, puis à se connecter à un serveur HTTP distant (ou à un autre proxy), qui se connecte à son tour au serveur MQIPT distant. Ce MQIPT côté serveur de la connexion n'a pas besoin de configuration spécifique, car la demande de connexion est traitée comme une connexion HTTP normale.

MQIPT utilise les propriétés **HTTProxy** et **HTTPServer** pour distinguer les proxys locaux et distants. La route MQIPT avec l'ensemble de propriétés **HTTProxy** est considérée comme étant le proxy HTTP local, et la route MQIPT avec l'ensemble de propriétés **HTTPServer** est le serveur distant (ou proxy).

Les connexions HTTPS sont normalement établies à l'adresse de port d'écoute 443 sur le proxy / serveur HTTP, mais les propriétés **HTTProxyPort** et **HTTPServerPort** peuvent être utilisées pour remplacer cette valeur par défaut.

Prise en charge de SOCKS dans MQIPT

Un proxy SOCKS est un service réseau utilisé comme point d'exit contrôlé via un pare-feu. Une application activée pour SOCKS, s'exécutant à l'intérieur du pare-feu, peut utiliser le proxy SOCKS pour se connecter à une application distante.

MQIPT peut agir en tant que proxy SOCKS en activant la propriété **SocksServer**. Ceci permet à une application IBM MQ activée pour SOCKS de se connecter via MQIPT à un gestionnaire de files d'attente IBM MQ distant. Lors de l'utilisation de cette fonction, la destination cible et l'adresse du port de destination sont obtenues lors du processus d'établissement de liaison SOCKS et, par conséquent, les propriétés de route **Destination** et **DestinationPort** sont remplacées. Il s'agit d'une caractéristique importante pour la prise en charge de la mise en cluster d'IBM MQ.

MQIPT peut également agir en tant que client SOCKS, pour le compte d'une application IBM MQ locale non activée pour SOCKS. Cela est utile lorsque vous utilisez un pare-feu autorisant les connexions sortantes uniquement via un proxy SOCKS. Chaque route MQIPT peut être configurée pour communiquer avec un autre proxy SOCKS.

Voir [Configuration d'un proxy SOCKS](#) pour un exemple d'utilisation de SOCKS.

Mise en cluster dans MQIPT

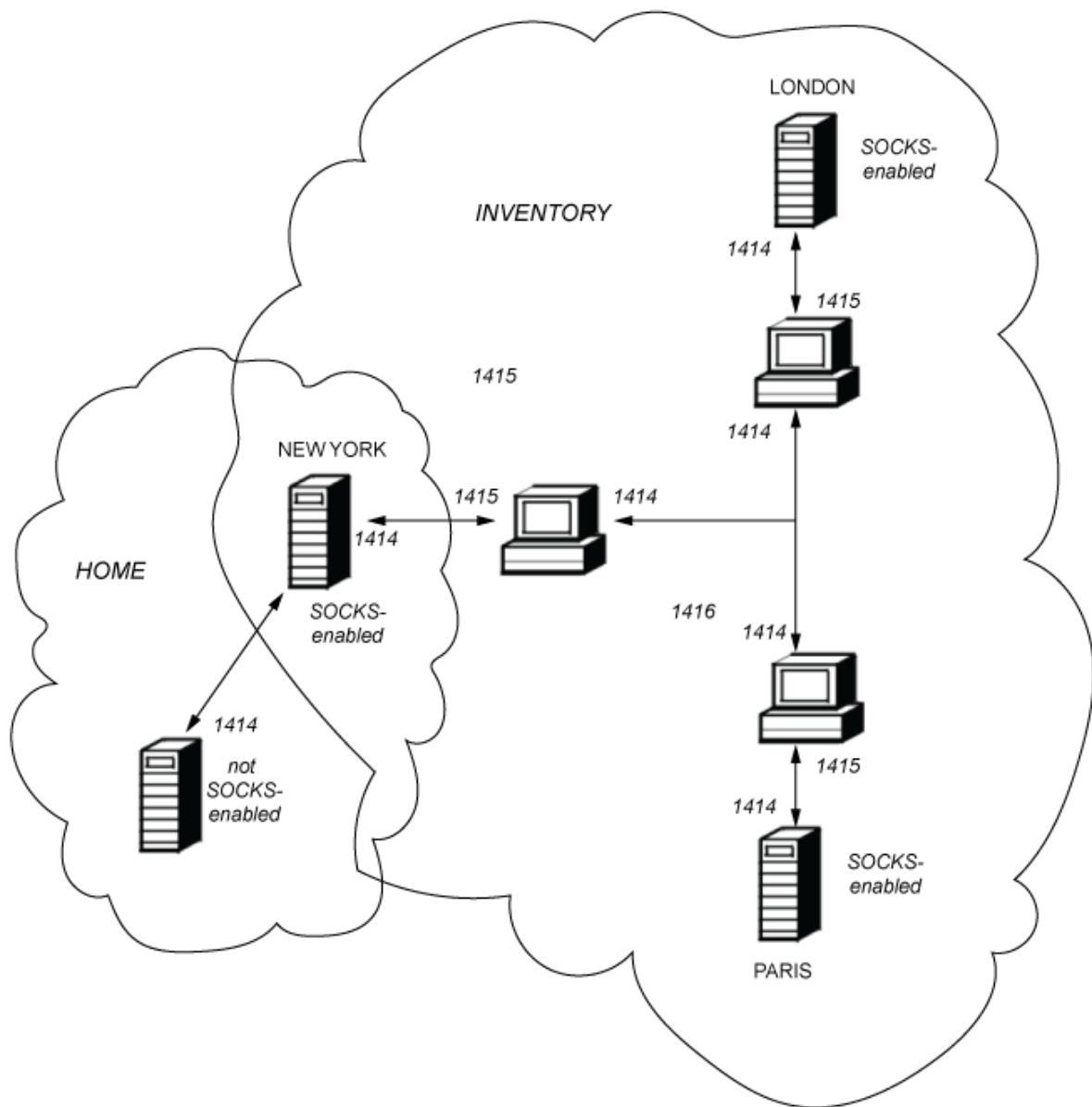
Les clusters IBM MQ peuvent être utilisés avec MQIPT en activant SOCKS pour chaque gestionnaire de files d'attente dans le cluster actif sur Internet et en configurant MQIPT afin qu'il agisse en tant que proxy SOCKS.

Dans le diagramme suivant, NEWYORK et CHICAGO se trouvent dans un cluster appelé HOME et les deux hébergent des référentiels complets. NEWYORK, LONDON et PARIS sont dans un autre cluster appelé INVENTORY. Notez que CHICAGO n'a pas besoin d'être activé pour SOCKS car il se trouve dans un cluster qui ne nécessite pas de MQIPT.

Chaque gestionnaire de files d'attente dans le cluster INVENTORY est effectivement "masqué" derrière un MQIPT. Comme le gestionnaire de files d'attente a été activé pour SOCKS, lorsqu'un canal émetteur de cluster est démarré, la demande est envoyée à sa destination, à l'aide de MQIPT qui agit en tant que proxy SOCKS. Normalement, le nom de connexion CONNAME sur un canal récepteur de cluster est utilisé pour identifier le gestionnaire de files d'attente local. Par contre, lorsqu'il est utilisé avec MQIPT, CONNAME doit identifier le MQIPT local et son port d'écoute entrant. Dans le diagramme suivant, toutes les adresses de port d'écoute entrant sont 1414 et les adresses de port de programme d'écoute sortant sont 1415.

Il existe deux façons d'exécuter un gestionnaire de files d'attente activé pour SOCKS. La première consiste à activer SOCKS pour la totalité de l'ordinateur sur lequel s'exécute le gestionnaire de files d'attente. La seconde consiste à activer SOCKS uniquement pour le gestionnaire de files d'attente. À l'aide de l'une ou l'autre méthode, vous devez configurer le client SOCKS uniquement afin qu'il effectue des connexions distantes avec MQIPT en tant que proxy SOCKS. Vous devez aussi désactiver l'authentification utilisateur. Plusieurs solutions du marché prennent en charge SOCKS. Vous devez choisir un produit compatible avec le protocole SOCKS V5.

Voir [Configuration de la prise en charge de la mise en cluster MQIPT](#) pour un exemple de configuration d'un réseau de cluster.



Prise en charge de SSL/TLS dans MQIPT

Les connexions sécurisées garantissent la confidentialité, l'intégrité et l'authentification des communications.

Confidentialité des communications

La connexion peut être protégée et définie comme privée. Les données devant être échangées entre le client et le serveur peuvent être chiffrées, de sorte que seul l'expéditeur et le destinataire puissent comprendre les données. Des informations privées, telles que des numéros de carte de crédit, peuvent ainsi être transférées en toute sécurité.

Intégrité des communications

La connexion est fiable. Le transport de messages inclut un contrôle d'intégrité de message utilisant une fonction de hachage sécurisé.

Authentification

Le client peut authentifier le serveur et un serveur authentifié peut authentifier le client. Cela garantit que les informations sont échangées uniquement entre les parties concernées. Le mécanisme d'authentification est basé sur l'échange de certificats numériques (certificats X.509v3).

Protocoles de connexion sécurisée

Dans MQIPT, les sockets sécurisés sont fournis à l'aide des protocoles TLS (Transport Layer Security) et SSL (Secure Sockets Layer). Ces deux protocoles sont similaires mais ne sont pas interopérables. Dans cette documentation, les termes SSL et TLS sont utilisés de manière interchangeable, sauf si une différence spécifique est notée.

MQIPT prend en charge SSL 3.0, TLS 1.0, TLS 1.1 et TLS 1.2 fournis par le Java runtime environment (JRE) fourni. Depuis IBM MQ 9.3.0, MQIPT prend également en charge TLS 1.3. IBM MQ CipherSpec du canal distant détermine les protocoles utilisés par MQIPT.

SSL 3.0, TLS 1.0 et TLS 1.1 ne sont pas sécurisés et sont désactivés par défaut dans MQIPT. Si vous devez utiliser l'un de ces protocoles désactivés, vous pouvez les réactiver en suivant la procédure décrite dans «[Activation de protocoles et de suites de chiffrement obsolètes dans MQIPT](#)», à la page 1093.

Les protocoles SSL/TLS utilisent des algorithmes de signature numérique différents pour authentifier les partenaires de la communication. Les opérations de chiffrement utilisées dans SSL/TLS, le chiffrement garantissant la confidentialité des données et le hachage sécurisé assurant l'intégrité des messages s'appuient sur le partage de clés secrètes entre le client et le serveur. Le protocole SSL/TLS fournit plusieurs mécanismes d'échange de ces clés secrètes. Il peut utiliser divers algorithmes de chiffrement et de hachage.

Activation du mode FIPS dans MQIPT

Le composant cryptographique SSL/TLS de l'environnement d'exécution Java contient le fournisseur de sécurité IBMJCEPlusFIPS, qui est certifié conforme à la norme FIPS 140-2. Si vous souhaitez utiliser uniquement la cryptographie certifiée FIPS dans MQIPT, activez le mode FIPS dans le fournisseur IBMJSSE2 en définissant les propriétés système Java suivantes lorsque MQIPT est démarré:

- `com.ibm.jsse2.usefipsprovider=true`
- `com.ibm.jsse2.usefipsProviderName=IBMJCEPlusFIPS`

Vous pouvez définir des propriétés système Java lorsque MQIPT est démarré à l'aide de la variable d'environnement **MQIPT_JVM_OPTIONS**. Par exemple, sous Linux, exécutez la commande suivante pour définir la variable d'environnement, avant d'exécuter la commande permettant de démarrer MQIPT:

```
export MQIPT_JVM_OPTIONS="-Dcom.ibm.jsse2.usefipsprovider=true  
-Dcom.ibm.jsse2.usefipsProviderName=IBMJCEPlusFIPS"
```

Pour plus d'informations sur l'activation du mode FIPS, voir [Activation du mode FIPS dans le fournisseur IBMJSSE2](#).

Mode de pontage SSL/TLS

Lorsque SSLServer et SSLClient sont définis pour une route, MQIPT accepte une connexion sécurisée SSL/TLS entrante et établit une deuxième connexion sécurisée SSL/TLS à un autre MQIPT ou à un gestionnaire de files d'attente de destination. Les informations du canal IBM MQ sont déchiffrées et rechiffrées entre ces deux connexions SSL/TLS. Le pontage SSL/TLS est également appelé *proxy de terminaison SSL/TLS*.

IBM MQ prend en charge le pontage SSL/TLS à l'aide de MQIPT. D'autres proxys de terminaison SSL/TLS avec IBM MQ ont été observés comme causant des connexions rompues si le proxy combine ou reconstruit des enregistrements SSL/TLS de tailles différentes de ceux envoyés par IBM MQ. Cela est dû à une interaction entre la façon dont les gestionnaires de files d'attente allouent et gèrent la mémoire pour les données réseau IBM MQ entrantes et la façon dont les données réseau IBM MQ sont regroupées dans des enregistrements SSL/TLS.

Le MQIPT conserve le conditionnement des données réseau IBM MQ dans les enregistrements SSL/TLS sans les fractionner ou les combiner. Si d'autres ponts SSL/TLS ne conservent pas exactement les enregistrements SSL/TLS, ils peuvent entraîner l'échec des canaux IBM MQ avec des messages d'erreur:

```
AMQ9638: SSL communications error for channel
AMQ9208: Error on receive from host
```

Mode proxy SSL/TLS

Une route MQIPT peut être configurée en mode proxy SSL/TLS comme alternative au pontage SSL/TLS. Dans ce mode, la route achemine uniquement les données SSL/TLS entre deux noeuds finaux IBM MQ ; elle ne participe pas à l'établissement de liaison SSL/TLS et ne requiert aucun certificat numérique.

Vous pouvez utiliser le mode proxy SSL/TLS dans les cas où les canaux IBM MQ qui communiquent via MQIPT sont déjà configurés pour la communication SSL/TLS et que vous souhaitez utiliser MQIPT à d'autres fins, telles que le routage des connexions via des pare-feux ou la restriction de l'ensemble des connexions autorisées via un exit de sécurité. Lors de l'exécution en mode proxy SSL/TLS, MQIPT vérifie que les paquets SSL/TLS initiaux reçus d'une nouvelle connexion sont valides avant de transférer les paquets à la destination.

IBM MQ prend en charge le mode proxy SSL/TLS avec MQIPT ou tout autre proxy SSL/TLS

IBM MQ prise en charge de plusieurs certificats avec MQIPT

IBM MQ 8.0, et versions ultérieures, prend en charge l'utilisation de plusieurs certificats sur le même gestionnaire de files d'attente, à l'aide d'un libellé de certificat par canal, spécifié à l'aide de l'attribut **CERTLABL** dans la définition de canal. Les canaux entrants dans le gestionnaire de files d'attente (par exemple, la connexion serveur ou le récepteur) dépendent de la détection du nom de canal à l'aide de TLS Server Name Indication (SNI), afin de présenter le certificat correct du gestionnaire de files d'attente. Pour plus d'informations sur l'utilisation de plusieurs certificats sur un gestionnaire de files d'attente, voir [Comment IBM MQ fournit plusieurs certificats](#).

Si un canal se connecte au gestionnaire de files d'attente de destination via MQIPT et que **SSLServer** et **SSLClient** sont définis pour la route MQIPT, il existe deux sessions TLS distinctes entre les noeuds finaux. Dans les versions antérieures à IBM MQ 9.3.0, les données SNI ne transitent pas par l'interruption de session. Cela empêche l'utilisation d'un certificat par canal sur le gestionnaire de files d'attente de destination pour la connexion TLS entre MQIPT et le gestionnaire de files d'attente. Pour utiliser un certificat par canal sur le gestionnaire de files d'attente de destination, pour une connexion TLS qui passe par MQIPT à une version antérieure à IBM MQ 9.3.0, la route MQIPT doit utiliser le mode proxy SSL/TLS, qui transmet tous les flux de contrôle TLS intacts, y compris le nom SNI.

Depuis la IBM MQ 9.3.0, MQIPT peut être configuré à l'aide de la propriété de route [SSLClientOutboundSNI](#) pour définir le SNI pour les connexions TLS sur une valeur spécifique ou pour passer par le SNI reçu sur la connexion entrante à la route. Pour autoriser l'utilisation de certificats par canal sur un gestionnaire de files d'attente de destination, la route doit être configurée pour définir le nom de canal IBM MQ comme SNI ou pour passer par le SNI reçu sur la connexion entrante à la route. Si MQIPT est configuré pour passer par l'interface SNI, le gestionnaire de files d'attente ou le client qui se connecte à MQIPT doit définir l'interface SNI sur le nom de canal.

Les certificats utilisés pour les connexions TLS qui sont arrêtées ou initiées par MQIPT peuvent être configurés individuellement pour chaque route, par exemple à l'aide des propriétés de route **SSLServerSiteLabel** et **SSLClientSiteLabel**.

Suites de chiffrement prises en charge par MQIPT

Tableau 71, à la page 1071 indique les CipherSuites prises en charge par MQIPT et celles qui sont activées par défaut.

Par défaut, seul un sous-ensemble de suites de chiffrement est activé. CipherSuites basés sur plusieurs algorithmes considérés comme non sécurisés sont désactivés par l'environnement d'exécution Java. Si vous connaissez les risques potentiels mais que vous devez toujours utiliser l'une de ces CipherSuites,

vous pouvez ajouter la prise en charge d'une CipherSuite désactivée en suivant la procédure décrite dans «Activation de protocoles et de suites de chiffrement obsolètes dans MQIPT», à la page 1093.

<i>Tableau 71. CipherSuites que vous pouvez utiliser avec MQIPT</i>	
CipherSuite	Activé par défaut
CipherSuites pour TLS 1.3	
TLS_AES_128_GCM_SHA256	Oui
TLS_AES_256_GCM_SHA384	Oui
TLS_CHACHA20_POLY1305_SHA256	Oui
CipherSuites for SSL 3.0, TLS 1.0, TLS 1.1 et TLS 1.2	
SSL_DH_anon_EXPORT_WITH_DES40_CBC_SHA	
SSL_DH_anon_EXPORT_WITH_RC4_40_MD5	
SSL_DH_anon_WITH_3DES_EDE_CBC_SHA	
SSL_DH_anon_WITH_AES_128_CBC_SHA	
SSL_DH_anon_WITH_AES_128_CBC_SHA256	
SSL_DH_anon_WITH_AES_128_GCM_SHA256	
SSL_DH_anon_WITH_AES_256_CBC_SHA	
SSL_DH_anon_WITH_AES_256_CBC_SHA256	
SSL_DH_anon_WITH_AES_256_GCM_SHA384	
SSL_DH_anon_WITH_DES_CBC_SHA	
SSL_DH_anon_WITH_RC4_128_MD5	
SSL_DHE_DSS_EXPORT_WITH_DES40_CBC_SHA	
SSL_DHE_DSS_WITH_3DES_EDE_CBC_SHA	
SSL_DHE_DSS_WITH_AES_128_CBC_SHA	Oui
SSL_DHE_DSS_WITH_AES_128_CBC_SHA256	Oui
SSL_DHE_DSS_WITH_AES_128_GCM_SHA256	Oui
SSL_DHE_DSS_WITH_AES_256_CBC_SHA	Oui
SSL_DHE_DSS_WITH_AES_256_CBC_SHA256	Oui
SSL_DHE_DSS_WITH_AES_256_GCM_SHA384	Oui
SSL_DHE_DSS_WITH_DES_CBC_SHA	
SSL_DHE_DSS_WITH_RC4_128_SHA	
SSL_DHE_RSA_EXPORT_WITH_DES40_CBC_SHA	
SSL_DHE_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_DHE_RSA_WITH_AES_128_CBC_SHA	Oui
SSL_DHE_RSA_WITH_AES_128_CBC_SHA256	Oui
SSL_DHE_RSA_WITH_AES_128_GCM_SHA256	Oui
SSL_DHE_RSA_WITH_AES_256_CBC_SHA	Oui
SSL_DHE_RSA_WITH_AES_256_CBC_SHA256	Oui

Tableau 71. CipherSuites que vous pouvez utiliser avec MQIPT (suite)

CipherSuite	Activé par défaut
SSL_DHE_RSA_WITH_AES_256_GCM_SHA384	Oui
SSL_DHE_RSA_WITH_DES_CBC_SHA	
SSL_ECDH_anon_WITH_3DES_EDE_CBC_SHA	
SSL_ECDH_anon_WITH_AES_128_CBC_SHA	
SSL_ECDH_anon_WITH_AES_256_CBC_SHA	
SSL_ECDH_anon_WITH_NULL_SHA	
SSL_ECDH_anon_WITH_RC4_128_SHA	
SSL_ECDH_ECDSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDH_ECDSA_WITH_AES_128_CBC_SHA	Oui
SSL_ECDH_ECDSA_WITH_AES_128_CBC_SHA256	Oui
SSL_ECDH_ECDSA_WITH_AES_128_GCM_SHA256	Oui
SSL_ECDH_ECDSA_WITH_AES_256_CBC_SHA	Oui
SSL_ECDH_ECDSA_WITH_AES_256_CBC_SHA384	Oui
SSL_ECDH_ECDSA_WITH_AES_256_GCM_SHA384	Oui
SSL_ECDH_ECDSA_WITH_NULL_SHA	
SSL_ECDH_ECDSA_WITH_RC4_128_SHA	
SSL_ECDH_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDH_RSA_WITH_AES_128_CBC_SHA	Oui
SSL_ECDH_RSA_WITH_AES_128_CBC_SHA256	Oui
SSL_ECDH_RSA_WITH_AES_128_GCM_SHA256	Oui
SSL_ECDH_RSA_WITH_AES_256_CBC_SHA	Oui
SSL_ECDH_RSA_WITH_AES_256_CBC_SHA384	Oui
SSL_ECDH_RSA_WITH_AES_256_GCM_SHA384	Oui
SSL_ECDH_RSA_WITH_NULL_SHA	
SSL_ECDH_RSA_WITH_RC4_128_SHA	
SSL_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA	Oui
SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	Oui
SSL_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	Oui
SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA	Oui
SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	Oui
SSL_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	Oui
SSL_ECDHE_ECDSA_WITH_NULL_SHA	
SSL_ECDHE_ECDSA_WITH_RC4_128_SHA	

Tableau 71. CipherSuites que vous pouvez utiliser avec MQIPT (suite)

CipherSuite	Activé par défaut
SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_ECDHE_RSA_WITH_AES_128_CBC_SHA	Oui
SSL_ECDHE_RSA_WITH_AES_128_CBC_SHA256	Oui
SSL_ECDHE_RSA_WITH_AES_128_GCM_SHA256	Oui
SSL_ECDHE_RSA_WITH_AES_256_CBC_SHA	Oui
SSL_ECDHE_RSA_WITH_AES_256_CBC_SHA384	Oui
SSL_ECDHE_RSA_WITH_AES_256_GCM_SHA384	Oui
SSL_ECDHE_RSA_WITH_NULL_SHA	
SSL_ECDHE_RSA_WITH_RC4_128_SHA	
SSL_KRB5_EXPORT_WITH_DES_CBC_40_MD5	
SSL_KRB5_EXPORT_WITH_DES_CBC_40_SHA	
SSL_KRB5_EXPORT_WITH_RC4_40_MD5	
SSL_KRB5_EXPORT_WITH_RC4_40_SHA	
SSL_KRB5_WITH_3DES_EDE_CBC_MD5	
SSL_KRB5_WITH_3DES_EDE_CBC_SHA	
SSL_KRB5_WITH_DES_CBC_MD5	
SSL_KRB5_WITH_DES_CBC_SHA	
SSL_KRB5_WITH_RC4_128_MD5	
SSL_KRB5_WITH_RC4_128_SHA	
SSL_RSA_EXPORT_WITH_DES40_CBC_SHA	
SSL_RSA_EXPORT_WITH_RC4_40_MD5	
SSL_RSA_WITH_3DES_EDE_CBC_SHA	
SSL_RSA_WITH_AES_128_CBC_SHA	Oui
SSL_RSA_WITH_AES_128_CBC_SHA256	Oui
SSL_RSA_WITH_AES_128_GCM_SHA256	Oui
SSL_RSA_WITH_AES_256_CBC_SHA	Oui
SSL_RSA_WITH_AES_256_CBC_SHA256	Oui
SSL_RSA_WITH_AES_256_GCM_SHA384	Oui
SSL_RSA_WITH_DES_CBC_SHA	
SSL_RSA_WITH_NULL_MD5	
SSL_RSA_WITH_NULL_SHA	
SSL_RSA_WITH_NULL_SHA256	
SSL_RSA_WITH_RC4_128_MD5	Oui
SSL_RSA_WITH_RC4_128_SHA	

Tableau 71. CipherSuites que vous pouvez utiliser avec MQIPT (suite)	
CipherSuite	Activé par défaut
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	
TLS_ECDHE_ECDSA_WITH_CHACHA20_POLY1305_SHA256	
TLS_DHE_RSA_WITH_CHACHA20_POLY1305_SHA256	

CipherSpecs et MQIPT CipherSuites

Le [Tableau 72](#), à la [page 1075](#) présente la relation entre les CipherSpecs prises en charge par IBM MQ et les CipherSuites prises en charge par MQIPT. Le tableau indique également la version de protocole qu'IBM MQ attend avec chaque CipherSpec.

Lorsqu'un gestionnaire de files d'attente ou un client IBM MQ communique avec MQIPT, sauf si vous utilisez le mode de proxy SSL dans MQIPT, vérifiez que le CipherSpec utilisé par IBM MQ correspond au CipherSuite utilisé par MQIPT:

- Lorsque MQIPT agit en tant que serveur TLS et que IBM MQ se connecte en tant que client TLS, le CipherSpec utilisé par IBM MQ doit correspondre à un CipherSuite sélectionné dans la configuration de route MQIPT .
- Lorsque MQIPT agit en tant que client TLS et se connecte à un gestionnaire de files d'attente IBM MQ qui agit en tant que serveur TLS, MQIPT CipherSuite doit correspondre au CipherSpec défini sur le canal IBM MQ récepteur.

Un CipherSpec IBM MQ détermine de manière unique l'algorithme de chiffrement et la version du protocole de socket sécurisé à utiliser. Certains CipherSpecs IBM MQ ne diffèrent que par la version du protocole. Si l'un de ces CipherSpecs est utilisé, il ne suffit pas de spécifier uniquement CipherSuite dans la configuration MQIPT . L'établissement de liaison SSL/TLS négocie la version de protocole de sockets sécurisés la plus élevée prise en charge par les deux côtés, puis sélectionne une CipherSuite dans l'ensemble de chiffrements mutuellement activés.

Par exemple, une route SSLClient avec `SSLClientCipherSuites=SSL_RSA_WITH_3DES_EDE_CBC_SHA` peut négocier l'utilisation de `TLS_RSA_WITH_3DES_EDE_CBC_SHA` (TLS 1.0) ou de `TRIPLE_DES_SHA_US` (SSL 3.0) avec le gestionnaire de files d'attente éloignées. Il est également possible d'utiliser cette CipherSuite avec TLS 1.2, mais IBM MQ ne prend pas en charge cette CipherSuite avec TLS 1.2. Pour cette raison, les routes SSLClient sont particulièrement susceptibles de provoquer des erreurs AMQ9616 ou AMQ9631 au niveau du gestionnaire de files d'attente.

Pour éviter les erreurs de ce type sur les routes SSLClient, définissez la propriété de route **SSLClientProtocols** sur la valeur appropriée pour le CipherSpec prévu. Dans certains cas, il peut également être nécessaire de restreindre le protocole défini côté serveur à l'aide de la propriété de route **SSLServerProtocols**. Utilisez la version de protocole indiquée dans le tableau pour déterminer le paramètre approprié pour ces propriétés de route.

Ce problème touche en particulier les suites de chiffrement et CipherSpecs suivants pour les routes SSLClient :

- `SSL_RSA_WITH_3DES_EDE_CBC_SHA`, qui correspond à :
 - SSL 3.0: MQ CipherSpec `TRIPLE_DES_SHA_US`
 - TLS 1.0: MQ CipherSpec `TLS_RSA_WITH_3DES_EDE_CBC_SHA`
- `SSL_RSA_WITH_DES_CBC_SHA`, qui correspond à :
 - SSL 3.0: MQ CipherSpec `DES_SHA_EXPORT`
 - TLS 1.0: MQ CipherSpec `TLS_RSA_WITH_DES_CBC_SHA`
- `SSL_RSA_WITH_RC4_128_SHA`, qui correspond à :
 - SSL 3.0: MQ CipherSpec `RC4_SHA_US`

- TLS 1.2: MQ CipherSpec TLS_RSA_WITH_RC4_128_SHA256

Si vous utilisez une seule route MQIPT SSLClient pour tunneller plusieurs canaux IBM MQ qui utilisent des CipherSpecs différents, vérifiez que tous les canaux sont configurés avec un CipherSpec qui utilise la même version de protocole de sockets sécurisés et que vous définissez la propriété de route **SSLClientProtocols** sur ce protocole.

Pour plus d'informations sur les CipherSpecs d'IBM MQ, voir [Activation des CipherSpecs](#).

Tableau 72. IBM MQ CipherSpecs correspondant à MQIPT CipherSuites

IBM MQ CipherSpec	MQIPT CipherSuite	Version de protocole
DES_SHA_EXPORT	SSL_RSA_WITH_DES_CBC_SHA	SSLv3
DES_SHA_EXPORT1024	Non disponible	Non disponible
ECDHE_ECDSA_3DES_EDE_CBC_SHA256	SSL_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA	TLSv1.2
ECDHE_ECDSA_AES_128_CBC_SHA256	SSL_ECDHE_ECDSA_WITH_AES_128_CBC_SHA256	TLSv1.2
ECDHE_ECDSA_AES_128_GCM_SHA256	SSL_ECDHE_ECDSA_WITH_AES_128_GCM_SHA256	TLSv1.2
ECDHE_ECDSA_AES_256_CBC_SHA384	SSL_ECDHE_ECDSA_WITH_AES_256_CBC_SHA384	TLSv1.2
ECDHE_ECDSA_AES_256_GCM_SHA384	SSL_ECDHE_ECDSA_WITH_AES_256_GCM_SHA384	TLSv1.2
ECDHE_ECDSA_NULL_SHA256	SSL_ECDHE_ECDSA_WITH_NULL_SHA	TLSv1.2
ECDHE_ECDSA_RC4_128_SHA256	SSL_ECDHE_ECDSA_WITH_RC4_128_SHA	TLSv1.2
ECDHE_RSA_3DES_EDE_CBC_SHA256	SSL_ECDHE_RSA_WITH_3DES_EDE_CBC_SHA	TLSv1.2
ECDHE_RSA_AES_128_CBC_SHA256	SSL_ECDHE_RSA_WITH_AES_128_CBC_SHA256	TLSv1.2
ECDHE_RSA_AES_128_GCM_SHA256	SSL_ECDHE_RSA_WITH_AES_128_GCM_SHA256	TLSv1.2
ECDHE_RSA_AES_256_CBC_SHA384	SSL_ECDHE_RSA_WITH_AES_256_CBC_SHA384	TLSv1.2
ECDHE_RSA_AES_256_GCM_SHA384	SSL_ECDHE_RSA_WITH_AES_256_GCM_SHA384	TLSv1.2
ECDHE_RSA_NULL_SHA256	SSL_ECDHE_RSA_WITH_NULL_SHA	TLSv1.2
ECDHE_RSA_RC4_128_SHA256	SSL_ECDHE_RSA_WITH_RC4_128_SHA	TLSv1.2
NULL_MD5	SSL_RSA_WITH_NULL_MD5	SSLv3
NULL_SHA	SSL_RSA_WITH_NULL_SHA	SSLv3
RC2_MD5_EXPORT	Non disponible	Non disponible
RC4_56_SHA_EXPORT1024	Non disponible	Non disponible
RC4_MD5_EXPORT	SSL_RSA_EXPORT_WITH_RC4_40_MD5	SSLv3

Tableau 72. IBM MQ CipherSpecs correspondant à MQIPT CipherSuites (suite)

IBM MQ CipherSpec	MQIPT CipherSuite	Version de protocole
RC4_MD5_US	SSL_RSA_WITH_RC4_128_MD5	SSLv3
RC4_SHA_US	SSL_RSA_WITH_RC4_128_SHA	SSLv3
TLS_AES_128_GCM_SHA256	TLS_AES_128_GCM_SHA256	TLSv1.3
TLS_AES_256_GCM_SHA384	TLS_AES_256_GCM_SHA384	TLSv1.3
TLS_CHACHA20_POLY1305_SHA256	TLS_CHACHA20_POLY1305_SHA256	TLSv1.3
TLS_RSA_WITH_3DES_EDE_CBC_SHA	SSL_RSA_WITH_3DES_EDE_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_128_CBC_SHA	SSL_RSA_WITH_AES_128_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_128_CBC_SHA256	SSL_RSA_WITH_AES_128_CBC_SHA256	TLSv1.2
TLS_RSA_WITH_AES_128_GCM_SHA256	SSL_RSA_WITH_AES_128_GCM_SHA256	TLSv1.2
TLS_RSA_WITH_AES_256_CBC_SHA	SSL_RSA_WITH_AES_256_CBC_SHA	TLSv1
TLS_RSA_WITH_AES_256_CBC_SHA256	SSL_RSA_WITH_AES_256_CBC_SHA256	TLSv1.2
TLS_RSA_WITH_AES_256_GCM_SHA384	SSL_RSA_WITH_AES_256_GCM_SHA384	TLSv1.2
TLS_RSA_WITH_DES_CBC_SHA	SSL_RSA_WITH_DES_CBC_SHA	TLSv1
TLS_RSA_WITH_NULL_NULL	Non disponible	Non disponible
TLS_RSA_WITH_NULL_SHA256	SSL_RSA_WITH_NULL_SHA256	TLSv1.2
TLS_RSA_WITH_RC4_128_SHA256	SSL_RSA_WITH_RC4_128_SHA	TLSv1.2
TRIPLE_DES_SHA_US	SSL_RSA_WITH_3DES_EDE_CBC_SHA	SSLv3

Etablissement de liaison SSL/TLS dans MQIPT

Le protocole d'établissement de liaison SSL/TLS est appliqué lors de la demande initiale de connexion entre le client et le serveur SSL/TLS, lors de l'authentification et de la négociation des suites de chiffrement.

Toutes les suites de chiffrement SSL/TLS prises en charge (voir «Prise en charge de SSL/TLS dans MQIPT», à la page 1068), sauf les suites de chiffrement anonymes, nécessitent une authentification du serveur et permettent l'authentification du client. Le serveur peut être configuré pour demander l'authentification du client. Il est conseillé d'éviter d'utiliser les suites de chiffrement anonymes, car elles n'offrent aucune garantie sur l'identité de l'homologue distant. Une attaque potentielle (de type man-in-the-middle) peut ainsi intercepter les connexions SSL/TLS anonymes à votre insu. Utilisez des suites de chiffrement anonymes uniquement sur des réseaux internes et uniquement si vous êtes prêt à accepter le risque de d'une interception de données.

L'authentification homologue de communication dans SSL/TLS repose sur un chiffrement à clé publique et des certificats numériques X.509v3. Un site qui doit être authentifié dans le protocole SSL/TLS requiert une clé privée et un certificat numérique (qui contient la clé publique correspondante ainsi que les informations sur l'identité du site), et la durée de validité du certificat. Les certificats sont signés par une autorité de certification. Ce type de certificat est appelé certificats de signataire. Un certificat suivi d'un ou de plusieurs certificats de signataire forme une chaîne de certificats. Une chaîne de certificats se caractérise par le fait que, en commençant par le premier certificat (certificat de site), la signature

de chaque certificat de la chaîne peut être vérifiée à l'aide de la clé publique du certificat de signataire suivant.

Lorsqu'une connexion sécurisée nécessitant l'authentification du serveur est en cours d'établissement, le serveur envoie au client une chaîne de certificats afin de prouver son identité. Le client SSL/TLS poursuit l'établissement de la connexion au serveur uniquement s'il peut authentifier le serveur, par exemple en vérifiant la signature du certificat de site du serveur. Pour vérifier cette signature, le client SSL/TLS doit sécuriser le site du serveur lui-même ou au moins l'un des signataires dans la chaîne de certificats fournie par le serveur. Les certificats des sites sécurisés et les signataires doivent être conservés côté client pour effectuer cette vérification.

Le client SSL/TLS inspecte la chaîne de certificats du serveur, en commençant par le certificat du site. Le client considère que la signature du certificat de site est valide dans les cas suivants :

- Le certificat du site se trouve dans le référentiel du site sécurisé ou des certificats de signataire.
- Un certificat de signataire de la chaîne peut être validé en fonction de son référentiel de certificats de signataire sécurisés.

Dans ce dernier cas, le client SSL/TLS vérifie que la chaîne de certificats est signée correctement, en commençant par le certificat de signataire sécurisé et en continuant jusqu'au certificat de site du serveur. Le format et les dates de validité de chaque certificat impliqué dans ce processus sont également examinés. Si l'une de ces vérifications échoue, la connexion au serveur est refusée. Après avoir vérifié le certificat du serveur, le client utilise la clé publique imbriquée dans ce certificat au cours des étapes suivantes du protocole SSL/TLS. La connexion SSL/TLS peut uniquement être établie si le serveur possède effectivement la clé privée correspondante.

L'authentification de client suit la même procédure : si un serveur SSL/TLS requiert l'authentification client, le client envoie au serveur une chaîne de certificats afin de prouver son identité. Le serveur vérifie la chaîne en fonction du référentiel du site sécurisé et des certificats de signataire. Après avoir vérifié le certificat du client, le serveur utilise la clé publique imbriquée dans ce certificat au cours des étapes suivantes du protocole SSL/TLS. La connexion SSL/TLS peut uniquement être établie si le client possède effectivement la clé privée correspondante.

Les versions récentes des protocoles TLS fournissent des communications de haute sécurité (SSL et les anciens protocoles TLS sont considérés comme non sécurisés). Cependant, le protocole agit en fonction des informations fournies par l'application. Seule une gestion sécurisée de ces informations garantit la sécurisation globale des communications. Par exemple, si votre référentiel d'un site sécurisé et des certificats de signataire est compromis, vous risquez d'établir une connexion sécurisée à un partenaire de communication qui n'est absolument pas sûr.

Implémentation de SSL/TLS dans MQIPT

SSL 3.0, TLS 1.0, TLS 1.1, TLS 1.2 et TLS 1.3 sont implémentés avec des jetons #12 PKCS (Public Key Cryptography Standards) stockés dans des fichiers de clés (avec les types de fichier .p12 ou .pfx), contenant X509.V3 V3. MQIPT peut également utiliser des magasins de clés de matériel de cryptographie qui prennent en charge la norme PKCS#11 Cryptographic Token Interface. MQIPT utilise le pack IBM Java Secure Socket Extension (JSSE).

MQIPT peut agir en tant que client ou serveur SSL/TLS, selon l'extrémité qui initialise la connexion. Le client lance une connexion et le serveur accepte la demande de connexion. Il est possible qu'une route MQIPT agisse à la fois en tant que client et en tant que serveur. Dans ce cas, l'utilisation de la fonction de mode proxy SSL/TLS offre généralement de meilleures performances.

Lorsque MQIPT est configuré pour le mode proxy SSL/TLS, il transmet uniquement les données SSL/TLS entre les deux noeuds finaux ; il ne participe pas à l'établissement de liaison SSL/TLS et ne requiert pas de certificats numériques.

Dans les versions antérieures à IBM MQ 9.3.0, MQIPT ne transmet pas les données TLS Server Name Indication (SNI) reçues sur une connexion TLS entrante via une connexion TLS sortante. Cela signifie que les certificats par canal, spécifiés à l'aide de l'attribut de canal **CERTLABL**, ne peuvent pas être utilisés pour les connexions TLS entre MQIPT et le gestionnaire de files d'attente de destination. Pour utiliser un certificat par canal sur le gestionnaire de files d'attente de destination, pour une connexion TLS qui

passer par MQIPT à une version antérieure à IBM MQ 9.3.0, la route MQIPT doit utiliser le mode proxy SSL/TLS, qui transmet tous les flux de contrôle TLS intacts, y compris le nom SNI. Depuis IBM MQ 9.3.0, MQIPT peut être configuré pour définir l'interface SNI pour les connexions TLS sur une valeur spécifique ou pour passer par l'interface SNI reçue sur la connexion entrante à la route. Pour plus d'informations sur l'utilisation de plusieurs certificats sur un gestionnaire de files d'attente avec MQIPT, voir [«IBM MQ prise en charge de plusieurs certificats avec MQIPT»](#), à la page 1070.

Chaque route MQIPT peut être configurée indépendamment avec son propre ensemble de propriétés SSL/TLS. Pour plus d'informations, voir [Propriétés de routeMQIPT](#).

Chiffrement d'un mot de passe de fichier de clés dans MQIPT

Les mots de passe utilisés par MQIPT pour ouvrir un fichier de clés ou pour accéder au matériel de cryptographie doivent être chiffrés à l'aide de la commande `mqiptPW`.

Le mot de passe chiffré peut être spécifié dans l'une des propriétés suivantes:

- `SSLClientKeyRingPW`
- `SSLClientCAKeyRingPW`
- `SSLServerKeyRingPW`
- `SSLServerCAKeyRingPW`
- `SSLCommandPortKeyRingPW`

Dans les versions antérieures à IBM MQ 9.1.5, les mots de passe de fichier de clés à utiliser par MQIPT sont stockés dans des fichiers référencés par les propriétés `SSL*KeyRingPW`.

Chiffrez les mots de passe de fichier de clés à utiliser par MQIPT à l'aide de la commande `mqiptPW` et définissez la valeur des propriétés `SSL*KeyRingPW` sur le mot de passe chiffré. MQIPT peut faire la distinction entre les mots de passe chiffrés et les noms de fichier dans les valeurs de propriété à des fins de compatibilité avec les configurations créées avant IBM MQ 9.1.5.

Deprecated La méthode de chiffrement des mots de passe de fichier de clés dans les versions de MQIPT antérieures à IBM MQ 9.1.5 est obsolète, mais elle peut toujours être utilisée. Pour améliorer la protection des mots de passe de fichier de clés, chiffrez à nouveau tous les mots de passe de fichier de clés chiffrés à l'aide de la méthode obsolète, à l'aide de la méthode de protection la plus récente.

Pour chiffrer un mot de passe de fichier de clés à utiliser par MQIPT, suivez les étapes de la rubrique [«Chiffrement des mots de passe stockés dans MQIPT»](#), à la page 1105.

Sélection de certificats à partir d'un fichier de clés dans MQIPT

Il est possible que plusieurs certificats personnels soient stockés dans le même fichier de clés ou jeton matériel de cryptographie. Les propriétés `SSLClientSite*` peuvent être utilisées côté client pour sélectionner le certificat à envoyer au serveur pour l'authentification et les propriétés `SSLServerSite*` peuvent être utilisées côté serveur pour sélectionner le certificat à envoyer au client pour l'authentification.

A l'aide de ces propriétés, un certificat peut être sélectionné en fonction de son nom distinctif (DN). Une autre solution consiste à utiliser le label de certificat pour sélectionner un certificat à l'aide des propriétés `SSLServerSiteLabel` et `SSLClientSiteLabel`.

Pour sélectionner le certificat serveur utilisé par le port de commande TLS, utilisez la propriété `SSLCommandPortSiteLabel` pour spécifier le nom de libellé du certificat.

Paramètres de confiance dans MQIPT

Un magasin de clés contient un certificat personnel qui inclut le certificat de signataire ou la chaîne de certificats de signataire.

MQIPT utilise deux types de magasin de clés:

Magasin de clés de l'autorité de certification

Ce magasin de clés contient des certificats de l'autorité de certification de confiance qui sont utilisés pour valider les certificats appartenant à un homologue distant, et servent à déterminer si l'homologue distant est fiable. MQIPT prend en charge les magasins de clés au format PKCS #12 et les magasins de clés de matériel de cryptographie qui prennent en charge l'interface PKCS #11 pour le stockage des certificats de l'autorité de certification.

Les magasins de clés de l'autorité de certification MQIPT sont identifiés par les propriétés de route **SSLClientCAKeyRing** et **SSLServerCAKeyRing**. L'utilisation du matériel de cryptographie pour accéder aux certificats de l'autorité de certification est activée en définissant les propriétés **SSLClientCAKeyRingUseCryptoHardware** et **SSLServerCAKeyRingUseCryptoHardware**.

Le magasin de clés de l'autorité de certification côté client SSL/TLS doit contenir les certificats de l'autorité de certification de confiance utilisés pour authentifier le certificat envoyé par le serveur. Si une route de serveur SSL est configurée pour l'authentification du client, le magasin de clés de l'autorité de certification côté serveur SSL/TLS doit contenir les certificats de l'autorité de certification de confiance utilisés pour authentifier le certificat envoyé par le client.

Magasin de clés de certificat personnel

Ce magasin de clés contient des certificats personnels que MQIPT utilise pour s'identifier auprès d'un homologue distant. Lorsque vous générez un certificat autosigné ou que vous demandez un certificat signé par une autorité de certification, utilisez le magasin de clés de certificat personnel.

MQIPT prend en charge les magasins de clés au format PKCS #12 et les magasins de clés de matériel de cryptographie qui prennent en charge l'interface PKCS #11 pour le stockage des certificats personnels.

Les magasins de clés de certificats personnels sont identifiés par les propriétés de route **SSLClientKeyRing** et **SSLServerKeyRing**. L'utilisation du matériel de cryptographie pour accéder aux certificats personnels est activée en définissant les propriétés **SSLClientKeyRingUseCryptoHardware** et **SSLServerKeyRingUseCryptoHardware**.

Le magasin de clés côté serveur SSL/TLS doit contenir le certificat personnel du serveur MQIPT. Si l'authentification du client est nécessaire sur une route de client SSL, le magasin de clés côté client SSL/TLS doit contenir le certificat personnel du client.



Si vous avez besoin d'une authentification client, vous devez activer la propriété

SSLServerAskClientAuth côté serveur. Le magasin de clés côté client doit contenir le certificat personnel du client. Le magasin de clés MQIPT côté serveur, qui est identifié par la propriété **SSLServerCAKeyRing**, doit contenir les certificats de l'autorité de certification de confiance utilisés pour authentifier le client.

Si vous ne configurez pas de magasin de clés de l'autorité de certification pour une route, MQIPT recherche à la place les certificats de l'autorité de certification dans le magasin de clés du certificat personnel, s'il en existe un qui est configuré. Par exemple, si aucune valeur n'est définie pour **SSLServerCAKeyRing**, MQIPT recherche les certificats de l'autorité de certification dans le magasin de clés identifié par **SSLServerKeyRing**.

Comme alternative aux certificats signés par une autorité de certification sécurisée, vous pouvez utiliser des certificats autosignés. Vous trouverez un exemple de certificat autosigné dans l'exemple de magasin de clés `sslSample.pfx` fourni avec MQIPT dans le sous-répertoire `samples/ssl`. Pour ouvrir les exemples de magasin de clés PKCS#12, vous devez utiliser le mot de passe `mqiptSample`.

Les certificats autosignés peuvent être utiles dans les scénarios de test au cours desquels vous devez vérifier la connectivité SSL/TLS sans acheter un certificat auprès d'une autorité de certification. Toutefois, n'utilisez pas de certificats autosignés dans les environnements de production. Pour créer un certificat signé par une autorité de certification, voir [Création d'un fichier de clés](#).

Vous pouvez utiliser la commande   **mqiptKeytool** pour gérer les certificats numériques et les magasins de clés. Pour plus d'informations, voir [«Gestion des magasins de clés MQIPT»](#), à la page 1083.

Protégez les fichiers de clés et les fichiers de mots de passe en utilisant les fonctions de sécurité du système d'exploitation pour empêcher tout accès non autorisé à ces fichiers.

Test de SSL/TLS dans MQIPT

Exemples pour vous aider à tester une connexion SSL/TLS.

Pour une description des différents scénarios, voir [Initiation à IBM MQ Internet Pass-Thru](#) . En particulier, consultez les tâches suivantes :

- [Authentification d'un serveur SSL/TLS](#)
- [Authentification d'un client SSL/TLS](#)
- [Exécution de MQIPT en mode proxy SSL/TLS](#)
- [Exécution de MQIPT en mode proxy SSL/TLS avec un gestionnaire de sécurité](#)

Pour vérifier que votre configuration SSL/TLS fonctionne correctement, vous pouvez utiliser des certificats autosignés. Les certificats autosignés sont utiles dans les scénarios de test pour que vous puissiez tester la connectivité SSL/TLS sans avoir à payer une autorité de certification pour un certificat. Pour plus d'informations, voir [Création de certificats de test](#).

N'utilisez pas de certificats autosignés dans les environnements de production. Au lieu de cela, procurez-vous un certificat signé par une autorité de certification sécurisée. Pour créer un certificat signé par une autorité de certification, voir [Création d'un fichier de clés](#).

Lorsque vous créez ou demandez un certificat, vous devez prendre en compte le type de clé, la taille de clé et l'algorithme de signature numérique adaptés à vos besoins de sécurité. Pour plus d'informations, voir «[Considérations relatives aux certificats numériques dans MQIPT](#)», à la page 1085.

Les certificats et les technologies de gestion des certificats sont disponibles auprès de plusieurs fournisseurs tiers.

Exemples de fichiers de clés MQIPT

Les exemples de fichier de clés PKCS #12 suivants sont fournis avec MQIPT dans le sous-répertoire `samples/ssl` à des fins de commodité lors des tests:

sslSample.pfx

Fichier de clés contenant un exemple de certificat autosigné.

sslCASample.pfx

Exemple de fichier de clés de certificat de l'autorité de certification.

Pour accéder à ces exemples de fichier de clés, utilisez le mot de passe `mqiptSample`.

L'exemple de certificat autosigné doit être utilisé uniquement dans un environnement de test car les clés privées du certificat sont disponibles pour tous les utilisateurs MQIPT .

Messages d'erreur SSL/TLS dans MQIPT

Les échecs d'établissement de liaison sont consignés dans le journal des connexions MQIPT sous la forme d'exceptions JSSE

Pour plus d'informations, voir «[Journaux de connexion dans MQIPT](#)», à la page 1107. Le tableau suivant décrit les différentes exceptions, la cause probable et l'action correspondante pour résoudre l'incident.

Les exceptions de certificat se rapportent généralement aux certificats à l'extrémité distante de la connexion.

Lorsque l'erreur est liée au certificat d'un client ou d'un gestionnaire de files d'attente IBM MQ , le terme *fichier de clés* inclut le référentiel de clés IBM MQ du partenaire distant.


Dans MQIPT, les certificats de l'autorité de certification sont stockés dans le fichier de clés de l'autorité de certification, qui est identifié par les propriétés de route **SSLClientCAKeyRing** et **SSLServerCAKeyRing** . Si les propriétés de route du fichier de clés de l'autorité de certification ne sont pas définies, le fichier de clés personnel correspondant (référéncé par la propriété **SSLClientKeyRing** ou **SSLServerKeyRing**) est recherché à la place pour les certificats de l'autorité de certification.

Exception	Cause	Action
CertificateException	Le certificat n'est pas digne de confiance car il est signé par une autorité de certification qui ne se trouve pas dans le fichier de clés de l'autorité de certification.	Vérifiez que tous les certificats de l'autorité de certification nécessaires sont présents dans le fichier de clés de l'autorité de certification. Utilisez l'outil IBM Key Management fourni avec MQIPT pour ajouter les certificats d'autorité de certification manquants. Veillez à obtenir une copie de chaque certificat d'autorité de certification provenant d'une source fiable.
CertificateExpiredException	<ol style="list-style-type: none"> 1. Le certificat est arrivé à expiration : sa date notAfter est dépassée. 2. L'horloge système est réglée de manière incorrecte. 	<ol style="list-style-type: none"> 1. Obtenez un nouveau certificat et insérez-le dans le fichier de clés. Si le certificat appartient à une autorité de certification, placez le nouveau certificat dans le fichier de clés de l'autorité de certification. 2. Vérifiez que l'heure de l'horloge système de temps universel coordonné (UTC) est correcte.
CertificateNotYetValidException	<ol style="list-style-type: none"> 1. L'utilisation du certificat est prématurée : sa date notBefore n'est pas encore échu. 2. L'horloge système est réglée de manière incorrecte. 	<ol style="list-style-type: none"> 1. Vérifiez que le certificat a été généré et signé correctement. Si votre organisation gère sa propre autorité de certification, l'horloge système de temps universel coordonné (UTC) de l'autorité de certification est peut-être mal réglée. 2. Vérifiez que l'heure de l'horloge système de temps universel coordonné (UTC) est correcte.
CertificateParsingException	<ol style="list-style-type: none"> 1. Le certificat contient des données DER non valides. 2. Le certificat utilise des fonctions DER non prises en charge. 	Assurez-vous que le certificat a été correctement généré et peut être affiché dans l'outil IBM Key Management fourni avec MQIPT. Procurez-vous un nouveau certificat avec moins d'extensions de certificat.
CertificateRevokedException	La vérification de révocation de certificat est activée et a détecté que le certificat a été révoqué.	Le certificat en question n'est pas sécurisé. Obtenez un certificat de remplacement et vérifiez que le nouveau certificat et sa clé privée sont présents dans le fichier de clés.

Exception	Cause	Action
CertPathBuilderException	La chaîne de certificats n'a pas été signée par une autorité de certification reconnue.	<ol style="list-style-type: none"> 1. Si vous utilisez des certificats signés par une autorité de certification, vérifiez que tous les certificats de l'autorité de certification racine et intermédiaire sont présents dans le fichier de clés de l'autorité de certification. 2. Si vous utilisez des certificats autosignés, vérifiez que vous avez extrait une copie de la partie publique du certificat distant et que vous l'avez ajoutée au fichier de clés de l'autorité de certification. Evitez d'utiliser des certificats autosignés dans les environnements de production.
CertStoreException KeyStoreException	<p>Une erreur s'est produite lors de la lecture d'un certificat à partir d'un fichier de clés pour l'une des raisons suivantes:</p> <ol style="list-style-type: none"> 1. Le fichier de clés est endommagé. 2. Le fichier de clés est manquant. 3. Le mot de passe stocké ne correspond pas au mot de passe du fichier de clés. 4. Si la route est configurée pour utiliser du matériel de cryptographie, MQIPT n'a pas pu se connecter au matériel de cryptographie. 	<ol style="list-style-type: none"> 1. Vérifiez que le fichier de clés peut être lu et que tous les certificats peuvent être affichés à l'aide de l'outil IBM Key Management. 2. Vérifiez que toutes les propriétés de route du fichier de clés font référence au nom de fichier correct. 3. Vérifiez que le mot de passe du fichier de clés stocké est correct. Utilisez l'outil mqiptPW pour stocker le mot de passe correct. 4. Si la route est configurée pour utiliser du matériel de cryptographie, vérifiez les points suivants: <ul style="list-style-type: none"> • Le fichier de propriétés de sécurité Java indique que le fournisseur de sécurité IBMPKCS11Impl est installé. • Le fichier de propriétés de sécurité Java contient le nom qualifié complet du fichier de configuration utilisé pour initialiser le fournisseur de sécurité IBMPKCS11Impl . • Le fichier de configuration utilisé pour initialiser le fournisseur de sécurité IBMPKCS11Impl est valide.

Exception	Cause	Action
SSLException: Aucun certificat ou clé disponible ne correspond aux suites de chiffrement SSL qui sont activées.	Vous devez disposer d'un certificat personnel avec le type correct de clé pour les suites de chiffrement que vous utilisez. Par exemple, les suites de chiffrement dont les noms commencent par SSL_ECDH_ECDSA_ nécessitent un certificat avec une clé publique Elliptic Curve. Les suites de chiffrement les plus couramment utilisées nécessitent un certificat doté d'une clé publique RSA.	Ouvrez le fichier de clés à l'aide de l'outil IBM Key Management. Sous la vue des certificats personnels, sélectionnez chaque certificat tour à tour et affichez-les. Cliquez sur l'option d' affichage des détails et accédez à la section Subject Public Key pour voir le type de clé publique. Vérifiez ensuite que les propriétés de route MQIPT SSLClientCipherSuites et SSLServerCipherSuites pour savoir si les suites de chiffrement adéquates sont activées.
SSLException: Aucune suite de chiffrement en commun SSLHandshakeException: aucune suite de chiffrement en commun	L'établissement de liaison n'a pas pu choisir une suite de chiffrement, car il n'existe pas de chevauchement entre les ensembles de suites de chiffrement activés aux deux extrémités de la connexion. Une connexion IBM MQ sortante active un seul algorithme de cryptographie. Par conséquent, les routes MQIPT sont particulièrement exposées à cette erreur. Cette erreur peut également se produire lorsque les trois conditions suivantes sont remplies: <ul style="list-style-type: none"> • aucun CipherSuite n'est spécifié sur la route • aucun certificat de site approprié ne peut être trouvé dans le fichier de clés configuré pour la route • Les CipherSuites anonymes sont désactivées 	Vérifiez la liste des suites de chiffrement activées dans les propriétés de route MQIPT SSLClientCipherSuites et SSLServerCipherSuites . Vous pouvez aussi envisager d'activer des suites de chiffrement supplémentaires. Consultez le tableau pour déterminer les suites de chiffrement à activer pour chaque valeur CipherSpec du canal IBM MQ. Si aucune CipherSuite n'est spécifiée sur la route, vérifiez que les propriétés de la route du fichier de clés font référence au fichier de clés correct et que le fichier de clés contient un certificat personnel que MQIPT peut utiliser. Si la route est configurée pour utiliser du matériel de cryptographie, vérifiez que l'attribut tokenlabel dans le fichier de configuration utilisé pour initialiser le fournisseur de sécurité IBMPKCS11Impl spécifie le libellé de jeton d'unité de cryptographie correct.

Gestion des magasins de clés MQIPT

  La commande **mqiptykeytool** permet de gérer les certificats dans les magasins de clés utilisés par IBM MQ Internet Pass-Thru (MQIPT).

Depuis la IBM MQ 9.4.0, cette commande remplace la commande **mqiptykeycmd** qui est utilisée pour gérer les certificats dans les versions antérieures de MQIPT.

Format de magasin de clés requis pour MQIPT

V 9.4.0 **V 9.4.0** MQIPT prend en charge les magasins de clés qui utilisent le format de fichier PKCS #12 . Lorsque vous utilisez la commande **mqiptKeytool** pour gérer le magasin de clés MQIPT , spécifiez le paramètre **-storetype pkcs12** pour indiquer que le magasin de clés utilise le format PKCS #12 .

MQIPT peut également accéder aux certificats stockés dans du matériel de chiffrement qui prend en charge l'interface PKCS #11 . L'interface peut également être utilisée pour gérer les certificats sur le matériel PKCS #11 . Pour plus d'informations, voir [«Utilisation du matériel de cryptographie PKCS #11 dans MQIPT»](#), à la page 1093.

Chiffrement du mot de passe du magasin de clés pour MQIPT

Chiffrez le mot de passe du magasin de clés dans un format que MQIPT peut utiliser pour accéder au fichier. Pour plus d'informations, voir [«Chiffrement d'un mot de passe de fichier de clés dans MQIPT»](#), à la page 1078.

La fonction de fichier de dissimulation prise en charge par IBM MQ n'est pas prise en charge par MQIPT. Utilisez la commande **mqiptPW** pour chiffrer le mot de passe du magasin de clés au lieu d'utiliser un fichier de dissimulation.

Exemples

V 9.4.0 **V 9.4.0**

Les exemples suivants montrent comment la commande **mqiptKeytool** est utilisée pour gérer les certificats dans un magasin de clés MQIPT .

- La commande suivante crée un certificat personnel autosigné à des fins de test:

```
mqiptKeytool -genkeypair -keystore key.p12 -storetype pkcs12 -storepass password
              -alias mqipt -dname "CN=Test Certificate,OU=Sales,O=Example,C=US"
              -keyalg RSA -keysize 2048 -sigalg SHA256withRSA
```

Cette commande crée un certificat numérique avec une clé publique RSA de 2048 bits et une signature numérique utilisant RSA avec l'algorithme de hachage SHA-256. Le certificat et ses clés publiques et privées associées sont stockés dans un magasin de clés au format PKCS #12 nommé `key.p12`. Le fichier de clés est créé s'il n'existe pas.

Lorsque vous créez un certificat, choisissez un algorithme de chiffrement de clé publique, une taille de clé et un algorithme de signature numérique adaptés aux besoins de sécurité de votre organisation. Pour plus d'informations, voir [«Considérations relatives aux certificats numériques dans MQIPT»](#), à la page 1085.

Cet exemple utilise un certificat autosigné utilisable dans un contexte de test. Dans un environnement de production, utilisez à la place un certificat signé par une autorité de certification.

- La commande suivante crée une demande de certificat pour un certificat signé par une autorité de certification à utiliser à des fins de production:

```
mqiptKeytool -certreq -keystore key.p12 -storetype pkcs12 -storepass password
              -alias mqipt -file cert.req
```

La commande crée une demande de signature de certificat (CSR) au format PKCS #10 . La demande de signature de certificat peut être envoyée à une autorité de certification pour demander un certificat signé par une autorité de certification. La paire de clés publique et privée avec l'alias `mqipt` doit être créée avant d'émettre cette commande, en émettant la commande **mqiptKeytool -genkeypair** dans l'exemple précédent.

- La commande suivante reçoit le fichier de certificat personnel signé par l'autorité de certification nommé `cert.crt` dans le magasin de clés:

```
mqiptKeytool -importcert -keystore key.p12 -storetype pkcs12 -storepass password -file
cert.crt
```

Importez le certificat de l'autorité de certification qui a signé le certificat personnel dans le magasin de clés en exécutant la commande suivante:

```
mqiptykeytool -importcert -keystore key.p12 -storetype pkcs12 -storepass password  
-file ca.crt -alias rootCA
```



Considérations relatives aux certificats numériques dans MQIPT

Les points à prendre en compte incluent la taille de clé de certificat, la sélection d'un algorithme de signature numérique approprié pour le certificat, ainsi que la compatibilité du certificat numérique et de la CipherSuite .

Taille de clé des certificats dans MQIPT

La taille de la clé publique dépend de la stratégie de sécurité de votre organisation et de l'algorithme de chiffrement utilisé. En général, les grandes tailles de clé sont plus sécurisées. Le tableau ci-dessous répertorie les tailles de clé minimum que vous devez utiliser :



Algorithme	Taille de clé minimum (en bits)
Elliptic Curve	256
RSA	2048

  Indiquez la taille de clé de votre certificat à l'aide du paramètre **-keysize** lorsque vous créez un certificat à l'aide de la commande **mqiptykeytool** .

Sélection d'un algorithme de signature numérique adéquat



Pour empêcher la falsification des certificats numériques, il est important d'utiliser un algorithme de signature numérique puissant. Veillez donc à choisir avec soin l'algorithme approprié lorsque vous créez ou demandez un certificat.

Évitez d'utiliser d'anciens algorithmes de signature numérique basés sur MD5 ou SHA-1 car ces algorithmes ne sont pas considérés comme sécurisés. Si possible, utilisez un algorithme de signature numérique récent basé sur SHA-2, par exemple SHA-256 avec RSA (SHA256WithRSA).

  Spécifiez l'algorithme de signature à l'aide du paramètre **-sigalg** lorsque vous créez un certificat à l'aide de la commande **mqiptykeytool** .

Compatibilité des certificats numériques et des suites de chiffrement dans MQIPT

Les suites de chiffrement ne peuvent pas toutes être utilisées avec tous les certificats numériques. Il existe différents types de suite de chiffrement, regroupés en fonction du préfixe de leur nom. Chaque type de suite de chiffrement impose des restrictions différentes concernant le type de certificat numérique utilisable. Ces restrictions s'appliquent à toutes les connexions SSL/TLS de MQIPT, mais concernent particulièrement les utilisateurs de la cryptographie Elliptic Curve. Lors de l'établissement d'une liaison de socket sécurisée, MQIPT sélectionne automatiquement un certificat personnel pour s'identifier qui convient à la CipherSuite négociée. Dans la plupart des cas, MQIPT interopère automatiquement avec l'homologue distant. Toutefois, dans certains scénarios, il peut être nécessaire d'utiliser une suite de chiffrement MQIPT spécifique pour interagir avec un système IBM MQ distant.

  La commande **mqiptykeytool** peut être utilisée pour créer des certificats avec des clés publiques DSA, RSA et Elliptic Curve. Adressez-vous à votre autorité de certification pour savoir comment créer d'autres types de certificat.

Le type de certificat numérique à utiliser dépend du type de suite de chiffrement que vous utilisez :

- Les suites de chiffrement dont les noms commencent par SSL_ECDH_ECDSA_ et SSL_ECDHE_ECDSA_ requièrent un certificat numérique avec une clé publique Elliptic Curve.

- Les suites de chiffrement dont les noms contiennent *anon* sont anonymes. Elles ne nécessitent pas de certificat numérique pour identifier l'homologue distant. Ces suites de chiffrement accélèrent les tâches de gestion des certificats dans des réseaux où un autre moyen d'authentification est utilisé. Toutefois, en règle générale, évitez de les utiliser en raison de l'absence d'authentification.
- Les autres suites de chiffrement requièrent un certificat numérique avec une clé publique RSA.

Exit de certificat dans MQIPT

L'objectif d'un exit de certificat est de valider un certificat homologue SSL/TLS reçu par MQIPT.

Vous pouvez configurer une route MQIPT afin qu'elle agisse en tant que client SSL/TLS lorsqu'elle établit une nouvelle connexion et en tant que serveur SSL/TLS lorsqu'elle reçoit une demande de connexion. Pendant l'établissement de liaison SSL/TLS, un client SSL/TLS reçoit un certificat homologue du serveur, et le certificat est utilisé pour authentifier le serveur. Un serveur SSL/TLS peut également recevoir un certificat homologue sur le client, et le certificat est utilisé pour authentifier le client.

L'exit de certificat est appelé lorsque MQIPT reçoit un certificat homologue, vous permettant d'effectuer d'autres opérations de validation. Les exceptions interceptées par l'exit sont interceptées par MQIPT et la demande de connexion est interrompue. Par conséquent, il est recommandé que l'exit intercepte toutes les exceptions et renvoie un code retour approprié à MQIPT.

Un exemple est fourni pour montrer qu'un exit de certificat peut être implémenté pour plus d'informations, voir [Utilisation d'un exit de certificat pour authentifier un serveur SSL/TLS](#).

Remarque : MQIPT s'exécute dans un seul Java Virtual Machine de sorte qu'un exit de certificat défini par l'utilisateur peut compromettre le fonctionnement normal de MQIPT de l'une des manières suivantes:

- Impact sur les ressources système
- Génération de goulots d'étranglement
- Dégradation des performances

Vous devez tester les effets de votre exit de certificat de manière extensive avant de l'implémenter dans un environnement de production.

La classe com.ibm.mq.ipt.exit.CertificateExit dans MQIPT

La classe `com.ibm.mq.ipt.exit.CertificateExit` est une classe abstraite qui doit être implémentée par la classe définie avec la propriété `SSLExitName`.

La classe contient des implémentations par défaut pour l'exécution de l'exit et certaines méthodes publiques que vous pouvez éventuellement remplacer, en fonction de vos exigences. Liste complète des méthodes prises en charge :

Méthodes

public int init(IPTTrace)

La méthode `init` est appelée par MQIPT lorsque l'exit est chargé par MQIPT. Elle peut être implémentée pour effectuer toute initialisation de l'exit, par exemple, le chargement de données utilisées lors du processus de validation. L'implémentation par défaut n'effectue aucune opération

public int refresh(IPTTrace)

La méthode d'actualisation (`refresh`) est implémentée pour effectuer une actualisation des données, par exemple, le rechargement des données d'un disque utilisé lors du processus de validation. Cette méthode est appelée lorsque l'administrateur de MQIPT a émis une commande de régénération. L'implémentation par défaut n'effectue aucune opération

public void close(IPTTrace)

La méthode d'arrêt (`close`) est implémentée pour effectuer des opérations de nettoyage lorsque la route est sur le point d'être stoppée ou lorsque MQIPT est en cours d'arrêt. L'implémentation par défaut n'effectue aucune opération

public CertificateExitResponse validate(IPTTrace)

La méthode de validation (validate) est appelée pour effectuer la validation du certificat homologué. L'objet de retour peut être utilisé pour retransmettre des informations à MQIPT : par exemple, un code retour et un texte pouvant être ajoutés au journal de connexion. L'implémentation par défaut renvoie un CertificateExitResponse avec CertificateExitResponse.OK.

Méthodes d'obtention des propriétés prises en charge :

public int getListenerPort()

Extrait le port d'écoute de la route, défini par la propriété ListenerPort.

public String getDestination()

Extrait l'adresse de destination, définie par la propriété Destination.

public int getDestinationPort()

Extrait l'adresse du port d'écoute de destination, définie par la propriété DestinationPort.

public String getClientIPAddress()

Extrait l'adresse IP du client à l'origine de la demande de connexion.

public int getClientPortAddress()

Extrait l'adresse de port utilisée par le client à l'origine de la demande de connexion.

public boolean isSSLClient()

Détermine si l'exit est appelé en tant que client ou serveur SSL/TLS. Si la valeur renvoyée est true, l'exit se trouve du côté client de la connexion, et valide le certificat obtenu auprès du serveur. Si la valeur renvoyée est false, l'exit se trouve du côté serveur de la connexion, et valide le certificat envoyé par le client. Une route peut agir à la fois comme serveur SSL/TLS et comme client SSL/TLS, en déchiffrant et rechiffrant le trafic. Dans ce cas, bien qu'il existe une seule classe d'exit, certaines instances de la classe seront appelées en tant que clients et d'autres comme serveurs. Vous pouvez utiliser isSSLClient pour déterminer la situation d'une instance donnée.

public int getConnThreadID()

Extrait l'ID de l'unité d'exécution de tâche qui traite la demande de connexion, ce qui peut être utile pour le débogage.

public String getChannelName()

Extrait le nom du canal IBM MQ utilisé dans la demande de connexion. Cette option est disponible uniquement lorsque la demande entrante n'utilise pas SSL/TLS et que MQIPT agit en tant que client SSL/TLS.

public String getQMName()

Extrait le nom du gestionnaire de files d'attente IBM MQ utilisé dans la demande de connexion. Cette option est disponible uniquement lorsque la demande client n'utilise pas SSL/TLS et que MQIPT agit en tant que client SSL/TLS.

public boolean getTimeout()

Utilisé par l'exit pour déterminer si le délai d'attente a expiré.

public IPTCertificate getCertificate()

Extrait le certificat SSL/TLS à valider.

public String getExitData()

Extrait les données d'exit, définies par la propriété SSLExitData.

public String getExitName()

Extrait le nom de l'exit, défini par la propriété SSLExitName.

La classe com.ibm.mq.ipt.exit.CertificateExitResponse dans MQIPT

Cette classe permet de renvoyer des informations de MQIPT après la validation d'un certificat.

Constructeurs

public CertificateExitResponse(int rc, string message)

Ce constructeur permet de retourner un code retour et un texte de message. Les codes anomalie possibles sont les suivants :

- ExitRc.OK
- ExitRc.VALIDATE_ERROR
- ExitRc.VALIDATE_REJECTED

public CertificateExitResponse(int rc)

Ce constructeur permet de retourner un code retour sans texte de message. Les codes anomalie possibles sont les suivants :

- ExitRc.OK
- ExitRc.VALIDATE_ERROR
- ExitRc.VALIDATE_REJECTED

public CertificateExitResponse()

Ce constructeur permet de retourner un code retour ExitRc.OK, sans texte de message.

Méthodes

public String getVersion()

Cette méthode renvoie la version de cette classe.

public String toString()

Cette méthode renvoie une représentation de la réponse sous la forme d'une chaîne, par exemple, "Code anomalie : 4, Message : Echec de la vérification CRL."

La classe com.ibm.mq.ipt.exit.IPTCertificate dans MQIPT

Cette classe contient le certificat SSL/TLS à valider.

Méthodes

public int getVersion()

Cette méthode renvoie la version de cette classe.

public byte [] getDerEncoding()

Cette méthode renvoie le codage ASN.1/DER du certificat X.509, ou NULL si une erreur se produit.

public byte [] getPemEncoding()

Cette méthode renvoie le codage PEM (BASE64) du certificat X.509, ou NULL si une erreur se produit.

public String getLabel()

Cette méthode renvoie le label de certificat, ou NULL si une erreur se produit.

public String getName()

Cette méthode renvoie le nom distinctif du certificat, ou NULL si aucun nom n'est disponible.

Exemple :

```
CN=Test Queue Manager,OU=Sales,O=Example,L=London,C=GB
```

public String getIssuerName()

Cette méthode renvoie le nom distinctif de l'émetteur du certificat, ou NULL si aucun nom n'est disponible. Exemple :

```
CN=Certificate Authority,OU=Security,O=Example,L=New York,C=US
```


public IPTCertificate getSigner()

Cette méthode renvoie le certificat de signataire, ou NULL si aucun nom n'est disponible. Pour un certificat auto-signé, elle renvoie une référence à elle-même.

public String toString()

Cette méthode renvoie une représentation sous forme de chaîne du certificat.

La classe *com.ibm.mq.ipc.exit.IPTTrace* dans MQIPT

Les fonctions de trace MQIPT fournissent des appels d'entrée et de sortie, qui peuvent être utilisés à l'entrée et à la sortie d'une méthode. Il existe également divers appels de données pour tracer des informations utiles.

Méthodes

public void entry(String fid)

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode.

Cette méthode écrit une entrée dans le fichier de sortie de trace, avec le niveau de retrait approprié, pour enregistrer le point où le flux de contrôle arrive dans une méthode. Cet appel est facultatif, mais s'il est utilisé, un appel correspondant à "exit (String)" doit également être utilisé dans la même méthode.

public void exit(String fid)

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode.

Cette méthode écrit une entrée dans le fichier de sortie de trace, avec le niveau de retrait approprié, pour enregistrer le point où le flux de contrôle quitte une méthode. Cette méthode est utilisée uniquement lorsqu'un appel à "entrée (String)" a déjà été utilisé dans la même méthode.

public void exit(String fid, int rc)

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode, et *rc* est le code de retour numérique de la méthode. Cette méthode de trace doit être utilisée pour enregistrer l'exit à partir des méthodes renvoyant un entier.

Cette méthode écrit un exit dans le fichier de sortie de trace, avec le niveau de retrait approprié, pour enregistrer le point où le flux de contrôle quitte une méthode et pour noter le code de retour numérique de la méthode. Cette méthode est utilisée uniquement lorsqu'un appel à "entrée (String)" a déjà été utilisé dans la même méthode.

public void exit(String fid, boolean rc)

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode, et *rc* est le code de retour booléen de la méthode. Cette méthode de trace doit être utilisée pour enregistrer l'exit à partir des méthodes renvoyant un booléen.

Cette méthode écrit un exit dans le fichier de sortie de trace, avec le niveau de retrait approprié, pour enregistrer le point où le flux de contrôle quitte une méthode et pour noter le code de retour Booléen de la méthode. Cette méthode est utilisée uniquement lorsqu'un appel à "entrée (String)" a déjà été utilisé dans la même méthode.

public void data(String fid, String data)

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode.

Cette méthode permet d'écrire des données de type chaîne dans le fichier de sortie de trace.

public void data(String fid, int data)

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode.

Cette méthode permet d'écrire des données de type entier dans le fichier de sortie de trace.

public void data(String fid, byte[])

où *fid* est utilisé pour identifier l'origine de l'appel, par exemple le nom de classe et de méthode.

Cette méthode permet d'écrire des données binaires dans le fichier de sortie de trace.

Exemple de trace

Pour faciliter le diagnostic des problèmes dans un exit, vous pouvez utiliser les mêmes fonctions de traçage que MQIPT. Vous pouvez aussi implémenter vos propres fonctions de traçage. Si vous décidez d'utiliser les fonctions de traçage de MQIPT, il existe des appels d'entrée et d'exit qui peuvent être utilisés pour entrer dans une méthode et en sortir. Il existe également divers appels de données permettant de tracer des informations utiles comme illustré dans l'exemple suivant.

```
/**
 * This method is called to initialize the exit (for example, for
 * loading validation information) and place itself in a ready
 * state to validate connection requests.
 */
public int init(IPTTrace t) {
    final String fid = "MyExit.init";

    // Trace entry into this method
    t.entry(fid);

    // Trace useful information
    t.data(fid, "Starting exit - MQIPT version " + getVersion());

    // Perform initialization and load any data
    t.data(fid, "Ready for work");

    // Trace exit from this method
    t.exit(fid);

    return ExitRc.OK;
}
```

Cette méthode génère la trace dans le format indiqué dans l'exemple suivant :

```
16:36:48.625    14    5000-1s    -----{ ConnectionThread.setCertificateExit()
16:36:48.625    14    5000-1s    Creating instance of certificate exit
16:36:48.625    14    5000-1s    Calling init() of certificate exit
16:36:48.625    14    5000-1s    -----} MyExit.init()
16:36:48.625    14    5000-1s    Starting exit - MQIPT version 2.1.0.0
16:36:48.625    14    5000-1s    Ready for work
16:36:48.625    14    5000-1s    -----} MyExit.init() rc=0
16:36:48.625    14    5000-1s    -----} ConnectionThread.setCertificateExit() rc=0
```

Codes retour d'exit de certificat dans MQIPT

Les codes retour reconnus par MQIPT lors de l'appel d'un exit de certificat dans un certain nombre de situations différentes

Les codes retour suivants sont reconnus par MQIPT lors de l'appel d'un exit de certificat dans les situations suivantes :

Code retour	Description	init	Validat e	Actualis er
ExitRc.OK	L'exécution de la demande a abouti.	oui	oui	oui
ExitRc.INIT_ERROR	Echec de la demande d'initialisation, la route sera désactivée.	oui		
ExitRc.REFRESH_ERROR	Echec de la demande d'actualisation, la route sera désactivée.			oui
ExitRc.VALIDATE_ERROR	Le processus de validation a échoué, demande de connexion rejetée.		oui	

Code retour	Description	init	Validat e	Actualis er
ExitRc.VALIDATE_REJECTED	La demande de validation a échoué, demande de connexion rejetée.		oui	

LDAP et CRL dans MQIPT

MQIPT prend en charge un serveur LDAP (Lightweight Directory Access Protocol) qui exécute l'authentification des listes de révocation de certificats (CRL) sur un certificat numérique.

La prise en charge de LDAP a été mise en œuvre de la même façon que dans IBM MQ, car le même serveur LDAP peut être utilisé pour IBM MQ et MQIPT.

Au cours de l'établissement de liaison SSL/TLS, les partenaires de la communication s'authentifient mutuellement à l'aide de certificats numériques. L'authentification peut inclure une vérification du certificat reçu. Les autorités de certification (CA) révoquent les certificats pour diverses raisons, notamment :

- Le propriétaire est parti dans une autre entreprise.
- La clé privée n'est plus secrète.

Les autorités de certification publient les certificats personnels révoqués dans une liste de révocation de certificat (CRL). Les certificats d'autorités de certification qui ont été révoqués sont publiés dans une liste de révocation des droits d'accès (ARL). Les références ultérieures aux listes CRL s'appliquent également aux listes ARL.

Pour plus d'informations sur l'utilisation des serveurs LDAP avec IBM MQ et sur la gestion des listes de révocation de certificat et des listes de révocation de certificat, voir [Utilisation des listes de révocation de certificat et des listes de révocation de droit](#).

MQIPT peut prendre en charge jusqu'à deux serveurs LDAP sur chaque route. Le premier serveur LDAP est traité comme le serveur principal, le second est conservé en tant que serveur LDAP de secours. Le second serveur est utilisé uniquement si le serveur principal est inaccessible. Le serveur de sauvegarde doit être une image miroir du serveur principal.

L'accès aux informations stockées sur un serveur LDAP peut être protégé par un ID utilisateur et un mot de passe à l'aide des propriétés d'ID utilisateur et de mot de passe LDAP. Les mots de passe du serveur LDAP peuvent être chiffrés dans la configuration MQIPT à partir de IBM MQ 9.1.5. Pour plus d'informations sur le chiffrement des mots de passe à utiliser par MQIPT, voir [«Chiffrement des mots de passe stockés dans MQIPT»](#), à la page 1105.

Lorsque MQIPT charge un jeton PKCS #12 à partir d'un fichier de clés, la validité des certificats de l'autorité de certification est vérifiée. Si le certificat de l'autorité de certification est associée à une liste de révocation de certificat (CRL), il est vérifié afin de savoir s'il est arrivé à expiration. Le cas échéant, une nouvelle liste CRL est extraite du serveur LDAP. Toute liste CRL extraite est chargée dans le jeton en cours et rattachée à son certificat de l'autorité de certification.

Si aucune entrée ne correspond à l'autorité de certification lorsqu'une requête est envoyée au serveur LDAP principal, le système suppose qu'il n'existe aucune liste CRL pour cette autorité de certification et le serveur de sauvegarde n'est pas utilisé. Toutefois, si le serveur LDAP principal est inaccessible ou n'est pas rétabli au bout d'un laps de temps donné, le serveur de secours est utilisé. Les erreurs du serveur de secours mettent fin à la connexion client. Cette action peut être remplacée en définissant la propriété **LDAPIgnoreErrors** sur `true`.

Les listes CRL extraites par MQIPT sont conservées dans une mémoire cache et partagées par toutes les connexions sur cette route. Si une liste CRL mise en cache est arrivée à expiration, elle est supprimée de la mémoire cache et une nouvelle liste est extraite du serveur LDAP. Si une nouvelle liste CRL n'est pas disponible, la connexion est refusée.

L'arrivée à expiration d'une liste CRL extraite du serveur LDAP est également vérifiée. Un message d'avertissement est émis (MQCPW001). La liste CRL arrivée à expiration reste chargée dans le système

et toutes les demandes de connexion lui faisant référence sont refusées. Vous devez remplacer la CRL arrivée à expiration sur le serveur LDAP par une liste valide.

La propriété **LDAPCacheTimeout** peut être utilisée pour contrôler la fréquence de vidage du cache. La valeur par défaut est 1 jour. Une valeur de 0 signifie que les entrées de cache ne sont pas supprimées jusqu'à ce que la route ait été redémarrée.

Une CRL arrivée à expiration peut être stockée dans un fichier de clés ou sur un serveur LDAP. Si une nouvelle liste CRL n'a pas été émise, les demandes de connexion suivantes sont refusées. Vous pouvez ignorer les listes CRL arrivées à expiration en activant la propriété **IgnoreExpiredCRLs**.

Remarque : Si vous activez la propriété **LDAPIgnoreErrors** ou **IgnoreExpiredCRLs**, il se peut qu'un certificat révoqué soit utilisé pour établir une connexion SSL/TLS.

Propriétés d'unité organisationnelle de nom distinctif de certificat à plusieurs valeurs dans MQIPT

Vous pouvez faire correspondre plusieurs valeurs d'unités organisationnelles (OU) dans les noms distinctifs d'un certificat.

Les propriétés de route suivante prennent maintenant en charge de la mise en correspondance de plusieurs valeurs d'unités organisationnelles :

- **SSLClientDN_OU**
- **SSLClientSiteDN_OU**
- **SSLServerDN_OU**
- **SSLServerSiteDN_OU**

Pour faire correspondre plusieurs valeurs OU, utilisez une virgule comme séparateur dans la valeur de propriété de la route. Exemple :

```
SSLClientDN_OU=Sales, Europe
```

Cela fait correspondre les certificats avec OU=Sales et OU=Europe. Les valeurs OU sont mises en correspondance dans le même ordre que plusieurs valeurs d'unités organisationnelles dans des filtres SSLPEER d'IBM MQ.

N'indiquez pas la même propriété de route plusieurs fois dans la section [route]. La méthode appropriée pour faire correspondre plusieurs valeurs d'unités organisationnelles est de définir la propriété une seule fois, comme indiqué dans l'exemple précédent. Si vous entrez le même attribut plusieurs fois dans la même section mqipt.conf, c'est la dernière valeur qui prend effet. Par exemple, les entrées suivantes permettraient d'obtenir uniquement une correspondance avec Europe car la seconde ligne remplace la première :

```
SSLClientDN_OU=Sales  
SSLClientDN_OU=Europe
```

Si vous devez faire correspondre une virgule littérale dans une valeur OU, insérez une barre oblique inversée (\) comme caractère d'échappement immédiatement avant la virgule. Exemple :

```
SSLClientDN_OU=Sales\, Europe
```

Ceci met en correspondance une seule valeur : OU=Sales, Europe. Une barre oblique inversée qui n'est pas immédiatement suivie d'une virgule correspond à une barre oblique inversée littérale.

Si vous effectuez une mise à niveau d'une édition précédente de MQIPT et utilisez cette fonction pour mettre en correspondance des virgules dans des valeurs OU, vous devez insérer les caractères d'échappement de barre oblique inversée dans les propriétés de la route OU afin de conserver le comportement précédent.

Deprecated Activation de protocoles et de suites de chiffrement obsolètes dans MQIPT

Par défaut, les protocoles de sockets sécurisés et les suites de chiffrement considérés comme non sécurisés sont désactivés dans le Java runtime environment (JRE) fourni avec MQIPT. Ces protocoles et suites de chiffrement obsolètes doivent être activés avant de pouvoir être utilisés.

Pourquoi et quand exécuter cette tâche

Si vous connaissez les risques potentiels, mais que vous devez toujours utiliser l'un des protocoles ou suites de chiffrement considérés comme non sécurisés dans MQIPT, procédez comme suit pour activer le protocole ou la suite de chiffrement que vous devez utiliser.

Remarque : Les protocoles et les suites de chiffrement obsolètes ne peuvent pas être utilisés avec le port de commande TLS.

Procédure

1. Editez le fichier `java.security`, qui se trouve dans le répertoire `mqipt_path/java/jre/lib/security`, où `mqipt_path` est l'emplacement où MQIPT est installé.
2. Ajoutez la prise en charge de l'environnement d'exécution Java pour un protocole ou un algorithme en supprimant l'entrée correspondante de la liste des algorithmes désactivés dans la propriété `jdk.tls.disabledAlgorithms`.
 - Pour ajouter la prise en charge d'un protocole, supprimez le protocole de la liste des algorithmes désactivés. Par exemple, pour ajouter la prise en charge de TLS 1.0, supprimez `TLSv1` de la liste.
 - Pour ajouter la prise en charge d'une suite de chiffrement, supprimez les algorithmes correspondants de la liste des algorithmes désactivés. Par exemple, pour ajouter la prise en charge de la suite de chiffrement `SSL_ECDHE_ECDSA_WITH_3DES_EDE_CBC_SHA`, supprimez `3DES_EDE_CBC` et `DESede` de la liste.
3. Pour activer SSL 3.0 dans l'environnement d'exécution Java, vous devez également définir la propriété système `com.ibm.jsse2.disableSSLv3=false`.

Si vous démarrez MQIPT à partir de la ligne de commande à l'aide de la commande `mqipt`, vous pouvez définir la propriété à l'aide de la variable d'environnement `MQIPT_JVM_OPTIONS`. Exemple :

```
set MQIPT_JVM_OPTIONS=-Dcom.ibm.jsse2.disableSSLv3=false
```

Windows Si MQIPT est installé en tant que service Windows, vous pouvez définir la propriété en définissant une valeur de chaîne dans le registre Windows sous la clé `HKEY_LOCAL_MACHINE\SYSTEM\CurrentControlSet\Services\MQInternetPassThru`. La valeur doit avoir les attributs suivants:

Nom

`MqiptJvmOptions`

Données de la valeur

`-Dcom.ibm.jsse2.disableSSLv3=false`

4. Pour activer SSL 3.0, TLS 1.0 ou TLS 1.1 sur une route MQIPT, ajoutez le protocole correspondant à la propriété de route `SSLServerProtocols` ou `SSLClientProtocols`.
5. Redémarrez MQIPT pour que les modifications apportées aux propriétés de l'environnement d'exécution Java prennent effet.

Utilisation du matériel de cryptographie PKCS #11 dans MQIPT

MQIPT peut accéder aux certificats numériques stockés dans du matériel de cryptographie prenant en charge l'interface PKCS #11.

Avant de commencer

Avant de configurer MQIPT pour qu'il utilise du matériel de cryptographie, assurez-vous que la carte cryptographique, le pilote de carte et tout logiciel de support associé sont installés et fonctionnent correctement.

La prise en charge du matériel de cryptographie PKCS #11 dans MQIPT est assurée par le fournisseur de cryptographie IBM Java PKCS11 (fournisseur `IBMPKCS11Impl`). Pour plus d'informations sur le fournisseur `IBMPKCS11Impl` et la liste des cartes cryptographiques prises en charge par Java 8, voir [IBM PKCS11 Cryptographic Provider](#).

Pourquoi et quand exécuter cette tâche

Vous pouvez stocker les certificats personnels et les certificats de l'autorité de certification utilisés par MQIPT dans un magasin de clés matériel de cryptographie. Toutefois, comme une unité PKCS #11 ne dispose normalement pas de suffisamment d'espace pour stocker de nombreux certificats de signataire, vous pouvez utiliser un magasin de clés distinct basé sur des fichiers pour les certificats de l'autorité de certification.

Suivez cette procédure pour configurer MQIPT afin qu'il utilise des certificats dans un magasin de clés matériel de cryptographie.

Remarque :

- L'utilisation de matériel de cryptographie avec MQIPT est une fonction IBM MQ Advanced. Pour utiliser cette fonction, le gestionnaire de files d'attente local connecté à la route MQIPT doit également disposer des droits IBM MQ Advanced, IBM MQ Appliance, IBM MQ Advanced for z/OS VUE ou IBM MQ Advanced for z/OS.
-  La commande `mqiptKeytool` ne prend actuellement pas en charge le matériel de cryptographie PKCS #11. Cette procédure utilise la commande `ikeycmd` fournie avec Java runtime environment (JRE) à la place pour gérer les certificats dans le matériel de cryptographie. Assurez-vous que la commande `ikeycmd` de l'environnement d'exécution Java dans le chemin d'installation MQIPT est utilisée pour exécuter ces commandes.

Procédure

1. Créez le fichier de configuration utilisé lorsque le fournisseur `IBMPKCS11Impl` est initialisé.

Téléchargez des exemples de fichiers de configuration pour chacune des cartes cryptographiques matérielles prises en charge par le fournisseur `IBMPKCS11Impl` et configurez un exemple pour votre système. Les exemples peuvent être téléchargés à partir de la rubrique suivante dans IBM Documentation for Java: [Fichier de configuration](#).

Le fichier de configuration est un fichier texte et doit contenir au moins les attributs suivants:

name

Suffixe de nom de l'instance de fournisseur.

library

Nom qualifié complet de la bibliothèque PKCS #11 fournie avec le matériel de cryptographie.

tokenlabel

Libellé du jeton de l'unité de chiffrement PKCS #11.

Par exemple, le fichier de configuration peut contenir les entrées suivantes:

```
name = ITPKCS11Provider
library = /usr/lib64/pkcs11/PKCS11_API.so
tokenlabel = icatoken
```

2. Editez le fichier de propriétés de sécurité Java, `java.security`, situé dans le sous-répertoire `java/jre/lib/security` du répertoire d'installation MQIPT.
 - a) S'il n'est pas déjà présent dans le fichier, ajoutez le fournisseur de sécurité `IBMPKCS11Impl`. Par exemple, en ajoutant la ligne suivante:

```
security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl
```

- b) Ajoutez le nom qualifié complet du fichier de configuration après le nom du fournisseur. Par exemple, si le fichier de configuration que vous avez créé à l'étape «1», à la page 1094 s'appelle /opt/mqipt/pkcs11.cfg, ajoutez ce chemin à la même ligne que le fournisseur de sécurité:

```
security.provider.12=com.ibm.crypto.pkcs11impl.provider.IBMPKCS11Impl /opt/mqipt/  
pkcs11.cfg
```

3. Demandez un certificat personnel pour le matériel de cryptographie.

Entrez la commande suivante pour créer une demande de certificat à l'aide de la commande **ikeycmd** :

```
MQIPT_INSTALL_DIRECTORY/java/jre/bin/ikeycmd -certreq -create  
-crypto module_name -tokenlabel hardware_token  
-pw password -label label -size key_size  
-sig_alg algorithm -dn distinguished_name -file filename
```

où :

MQIPT_INSTALL_DIRECTORY

est le répertoire d'installation MQIPT.

-crypto nom_module

Indique le nom qualifié complet de la bibliothèque PKCS #11 fournie avec le matériel de cryptographie.

-tokenlabel libellé_jeton

Indique le libellé du jeton de l'unité de chiffrement PKCS #11 .

-pw mot_de_passe

Indique le mot de passe permettant d'accéder au matériel de cryptographie.

-label Libellé

Indique le libellé du certificat.

-size taille_clé

Indique la taille de la clé. La valeur peut être 512, 1024, 2048 ou 4096.

-sig_alg algorithm

Indique l'algorithme de signature asymétrique utilisé pour la création de la paire de clés de l'entrée. La valeur peut être MD2_WITH_RSA, MD2WithRSA, MD5_WITH_RSA, MD5WithRSA, SHA1WithDSA, SHA1WithECDSA, SHA1WithRSA, SHA2/ECDSA, SHA224WithECDSA, SHA256_WITH_RSA, SHA256WithECDSA, SHA256WithRSA, SHA2WithECDSA, SHA3/ECDSA, SHA384_WITH_RSA, SHA384WithECDSA, SHA384WithRSA, SHA3WithECDSA, SHA5/ECDSA, SHA512_WITH_RSA, SHA512WithECDSA, SHA512WithRSA, SHA5WithECDSA, SHA_WITH_DSA, SHA_WITH_RSA ou SHAWithDSA. La valeur par défaut est SHA256WithRSA.

-dn nom_distinctif

Indique le nom distinctif X.500 entre guillemets.

-file nom_fichier

Indique le nom de fichier de la demande de certificat.

4. Une fois que l'autorité de certification vous a envoyé le certificat personnel, ajoutez le certificat de l'autorité de certification au fichier de clés du matériel de cryptographie ou à un fichier de clés de l'autorité de certification, s'il n'est pas déjà présent.

- ▶ **V9.4.0** ▶ **V9.4.0** Pour ajouter le certificat de l'autorité de certification à un fichier de clés de l'autorité de certification PKCS #12 , entrez la commande suivante:

```
mqiptKeytool -importcert -keystore filename -storetype pkcs12 -storepass password  
-alias label -file cert_filename
```

où :

-keystore *nom_fichier*

Indique le nom du fichier de clés de l'autorité de certification. Le magasin de clés est créé s'il n'existe pas déjà.

-storepass *mot_de_passe*

Indique le mot de passe du magasin de clés d'autorité de certification

-alias *libellé*

Indique le libellé du certificat de l'autorité de certification.

-file *nom_fichier_certificat*

Indique le nom du fichier qui contient le certificat de l'autorité de certification.

- Pour ajouter le certificat de l'autorité de certification au matériel de cryptographie, entrez la commande suivante:

```
MQIPT_INSTALL_DIRECTORY/java/jre/bin/ikeycmd -cert -add  
-crypto module_name -tokenlabel hardware_token  
-pw password -label label -file cert_filename
```

où :

MQIPT_INSTALL_DIRECTORY

est le répertoire d'installation MQIPT.

-crypto *nom_module*

Indique le nom qualifié complet de la bibliothèque PKCS #11 fournie avec le matériel de cryptographie.

-tokenlabel *libellé_jeton*

Indique le libellé du jeton de l'unité de chiffrement PKCS #11 .

-pw *mot_de_passe*

Indique le mot de passe permettant d'accéder au matériel de cryptographie.

-label *Libellé*

Indique le libellé du certificat de l'autorité de certification.

-file *nom_fichier_certificat*

Indique le nom du fichier qui contient le certificat de l'autorité de certification.

5. Recevez le certificat personnel, fourni par l'autorité de certification, dans le magasin de clés du matériel de cryptographie.

Entrez la commande suivante pour ajouter le certificat au magasin de clés du matériel de cryptographie:

```
MQIPT_INSTALL_DIRECTORY/java/jre/bin/ikeycmd -cert -receive  
-crypto module_name -tokenlabel hardware_token  
-pw password -file filename
```

où :

MQIPT_INSTALL_DIRECTORY

est le répertoire d'installation MQIPT.

-crypto *nom_module*

Indique le nom qualifié complet de la bibliothèque PKCS #11 fournie avec le matériel de cryptographie.

-tokenlabel *libellé_jeton*

Indique le libellé du jeton de l'unité de chiffrement PKCS #11 .

-pw *mot_de_passe*

Indique le mot de passe permettant d'accéder au matériel de cryptographie.

-label *Libellé*

Indique le libellé du certificat.

-file *nom_fichier_certificat*

Indique le nom du fichier contenant le certificat à ajouter.

Si le certificat de l'autorité de certification est stocké dans un fichier de clés de l'autorité de certification et non dans le matériel de cryptographie, la commande émet un avertissement indiquant que la chaîne de certificats ne peut pas être validée car le fichier de clés de l'autorité de certification est inaccessible lorsque le certificat personnel est ajouté au fichier de clés matériel de cryptographie.

6. Chiffrez le mot de passe pour accéder au matériel de cryptographie à l'aide de la commande **mqiPTPW**.

Entrez la commande suivante :

```
mqiPTPW -sf encryption_key_file
```

où *encryption_key_file* est le nom d'un fichier qui contient la clé de chiffrement de mot de passe pour votre installation MQIPT. Vous n'avez pas besoin de spécifier le paramètre **-sf** si votre installation MQIPT utilise la clé de chiffrement de mot de passe par défaut. Entrez le mot de passe pour accéder au matériel cryptographique à chiffrer lorsque vous y êtes invité.

Pour plus d'informations sur le chiffrement des mots de passe de magasin de clés, voir «Chiffrement d'un mot de passe de fichier de clés dans MQIPT», à la page 1078.

7. Si vous avez ajouté le certificat de l'autorité de certification à un fichier de clés à l'étape «4», à la page 1095, suivez les instructions de l'étape «6», à la page 1097 pour chiffrer le mot de passe du fichier de clés de l'autorité de certification.

8. Editez le fichier de configuration `mqiPT.conf`.

- a) Confirmez que vous disposez des droits appropriés pour utiliser cette fonction IBM MQ Advanced en définissant la propriété globale **EnableAdvancedCapabilities** sur `true`.

- b) Activez l'utilisation du magasin de clés du matériel de cryptographie sur la route en définissant une ou plusieurs des propriétés **SSLServerKeyRingUseCryptoHardware**, **SSLServerCAKeyRingUseCryptoHardware**, **SSLServerKeyRingUseCryptoHardware** ou **SSLServerKeyRingUseCryptoHardware** sur `true`.

Pour plus d'informations sur les propriétés permettant d'activer l'utilisation du matériel de chiffrement sur une route, voir [Propriétés de la routeMQIPT](#).

Vous pouvez également utiliser du matériel de cryptographie avec le port de commande TLS en définissant la propriété **SSLCommandPortKeyRingUseCryptoHardware** sur `true`.

- c) Si vous utilisez un fichier de clés pour les certificats de l'autorité de certification, spécifiez l'emplacement du fichier de clés de l'autorité de certification en définissant une ou plusieurs des propriétés **SSLServerCAKeyRing** ou **SSLServerCAKeyRing**.

Si vous configurez la route pour utiliser du matériel de cryptographie pour le certificat de site et que vous ne spécifiez pas de fichier de clés de l'autorité de certification, le magasin de clés du matériel de cryptographie est utilisé comme magasin de clés de l'autorité de certification.

- d) Indiquez le mot de passe chiffré pour accéder au matériel de chiffrement et au magasin de clés de l'autorité de certification à l'aide de la propriété **SSLServerKeyRingPW**, **SSLServerCAKeyRingPW**, **SSLClientKeyRingPW**, **SSLClientCAKeyRingPW** ou **SSLCommandPortKeyRingPW**.

Définissez la valeur des propriétés **SSL*KeyRingPW** sur le mot de passe chiffré créé par la commande **mqiPTPW**.

- e) Si le matériel de cryptographie contient plusieurs certificats personnels, indiquez le certificat que MQIPT envoie au serveur ou au client SSL/TLS pour l'authentification.

- Pour une route client SSL/TLS, indiquez quel certificat est sélectionné à l'aide d'une ou de plusieurs des propriétés **SSLClientSite***.

- Pour une route de serveur SSL/TLS, indiquez quel certificat est sélectionné à l'aide d'une ou de plusieurs des propriétés **SSLServerSite***.

- Pour le port de commande TLS, indiquez le certificat sélectionné à l'aide de la propriété **SSLCommandPortSiteLabel** pour spécifier le libellé de certificat.

Pour plus d'informations sur la sélection de certificats dans un magasin de clés, voir «Sélection de certificats à partir d'un fichier de clés dans MQIPT», à la page 1078. Les propriétés permettant de sélectionner un certificat dans un magasin de clés sont décrites dans [Propriétés de routeMQIPT](#).


Par exemple, pour utiliser un magasin de clés matériel de cryptographie pour le certificat de site sur une route de serveur TLS et un fichier de clés pour stocker les certificats de l'autorité de certification pour la même route, ajoutez les propriétés suivantes à la définition de route:

```
SSLServerKeyRingUseCryptoHardware=true
SSLServerKeyRingPW=<mqiptPW>1!g0RdM4wft5d1rCgNMDEGag==!dZxhgQD2A8Ea0yeqawQvPg==
SSLServerCAKeyRing=/opt/mqipt/ssl/ca.pfx
SSLServerCAKeyRingPW=<mqiptPW>1!3Vdrpiu6kMwn0sWRCvGT5g==!LH1tGLEg30FvN8+02Re0YA==
SSLServerSitelabel=mqiptsite
```

9. Redémarrez MQIPT.

Java security manager dans MQIPT

Java security manager peut être utilisé avec toutes les fonctionsMQIPT afin de renforcer la sécurité.

Remarque :  L'utilisation de Java security manager avec MQIPT est obsolète car Java security manager a été déprécié en vue de sa suppression dans une édition ultérieure de Java.

MQIPT utilise le composant Java security manager par défaut défini dans la classe `java.lang.SecurityManager`. La fonction Java security manager de MQIPT peut être activée ou désactivée avec la propriété globale **SecurityManager**. Pour plus d'informations, voir [Propriétés globalesMQIPT](#).

Java security manager utilise deux fichiers de règles par défaut :

- Un fichier de règles système global nommé `$MQIPT_PATH/java/jre/lib/security/java.policy` (où `$MQIPT_PATH` est le répertoire dans lequel MQIPT est installé) est utilisé par toutes les instances d'une machine virtuelle sur un hôte.
- Un fichier de règles spécifiques à l'utilisateur appelé `.java.policy`, qui peut exister dans le répertoire de base de l'utilisateur.

Un fichier de règles MQIPT supplémentaire peut également être utilisé. Vous devez utiliser le fichier de règles MQIPT à la place des fichiers de règles par défaut décrits précédemment. Pour plus d'informations, voir **SecurityManagerPolicy** dans [Propriétés globales MQIPT](#).

La syntaxe du fichier de règles est très complexe. Il peut être modifié avec un éditeur de texte, mais il est généralement plus facile d'utiliser l'utilitaire Policy Tool fourni avec Java pour apporter des modifications. L'utilitaire Policy Tool se trouve dans le répertoire `$MQIPT_PATH/java/jre/bin` et il est entièrement documenté dans la documentation Java.

Un exemple de fichier de règle (`mqiptSample.policy`) a été fourni avec MQIPT. Il permet d'afficher les permissions à définir pour exécuter MQIPT.

Vous devez modifier le fichier d'exemple de règle afin qu'il corresponde à votre configuration. En particulier, notez que le répertoire de base MQIPT qui contient le fichier de configuration `mqipt.conf` peut ne pas être identique au répertoire d'installation MQIPT. Par conséquent, prenez soin de spécifier les répertoires appropriés lors de la configuration des entrées **FilePermission** dans la règle de sécurité.

Vous devez modifier les entrées suivantes :

- L'entrée **java.io.FilePermission** qui accorde à un accès en lecture et en écriture au répertoire `errors`. Le chemin d'accès au fichier dans cette entrée doit faire référence au répertoire de base MQIPT, car il s'agit de l'emplacement du répertoire `errors`. MQIPT crée des fichiers FFST Failure Data Capture (AMQ*.FDC) et des fichiers de trace (AMQ*.TRC*) dans le répertoire `errors`. Vous devez vous assurer que MQIPT est autorisé à créer des fichiers de trace et des fichiers FFST dans le répertoire `errors`, afin que le traitement des incidents soit possible.
- L'entrée **java.io.FilePermission** qui accorde à un accès en lecture et en écriture au répertoire `logs`. Le chemin d'accès au fichier dans cette entrée doit faire référence au répertoire de

base MQIPT , car il s'agit de l'emplacement du répertoire logs . MQIPT crée des fichiers journaux de connexion (mqipt*.log) dans le répertoire logs si la propriété globale **ConnectionLog** est activée.

- Les entrées **java.io.FilePermission** qui accordent à la lecture et à l'accès en exécution à tous les répertoires du répertoire d'installation MQIPT , tels que les répertoires bin, exits, libet ssl . Les chemins de fichier dans ces entrées doivent être modifiés afin de désigner le répertoire d'installation de MQIPT. Certaines de ces entrées peuvent être omises si elles ne sont pas obligatoires.
- Les entrées **java.net.SocketPermission** doivent être modifiées pour contrôler les connexions dans chaque route MQIPT à l'écoute. Les droits listen et accept sont requis pour le port d'écoute et l'adresse du programme d'écoute pour chaque route MQIPT .
- Les entrées **java.net.SocketPermission** doivent être modifiées pour contrôler les connexions sortantes de chaque route MQIPT. Le droit connect est requis pour les destinations de route, les serveurs proxy ou les serveurs LDAP auxquels la route MQIPT se connecte. Le droit resolve est requis lorsque vous spécifiez des destinations à l'aide d'un nom d'hôte au lieu d'une adresse IP.

Selon votre configuration, il peut être également nécessaire d'ajouter les entrées suivantes :

- Une entrée **java.io.FilePermission** pour accorder l'accès en lecture au fichier de configuration mqipt.conf ou au répertoire de base MQIPT contenant mqipt.conf.
- Une entrée **java.io.FilePermission** accordant le droit en lecture au fichier de règles de sécurité lui-même. Cela est pratique si une actualisation de MQIPT déclenche une nouvelle lecture du fichier de règles de sécurité.
- Certaines entrées **java.io.FilePermission** permettent d'accorder l'accès en lecture à des fichiers de clés SSL/TLS et à des fichiers de mot de passe de fichier de clés. Cette opération est requise uniquement lors de l'utilisation d'une route pour laquelle les propriétés **SSLClient** ou **SSLServer** sont activées ou lorsque le port de commande TLS est configuré.
- Certaines entrées **java.io.FilePermission** pour accorder à read ou execute l'accès à n'importe quelle classe d'exit MQIPT . Cela n'est obligatoire que lorsqu'un exit MQIPT est activé. Vous pouvez être amené à accorder des droits supplémentaires si l'exit le nécessite.

Remarque : Les entrées Windows **java.io.FilePermission** doivent utiliser deux barres obliques inversées (\\) pour chaque barre oblique inversée dans le chemin. En effet, une barre oblique inversée unique est utilisée comme caractère d'échappement.

L'exemple de fichier suppose que MQIPT a été installé sur un système Windows dans C : \Program Files\IBM\MQ Internet Pass-Thru. Il suppose également que le répertoire de base de MQIPT (l'emplacement du fichier mqipt.conf) est le même que le répertoire d'installation de MQIPT.

Si vous avez installé MQIPT dans un autre emplacement, vous devez modifier le répertoire dans la définition **codeBase** pour faire référence à votre répertoire d'installation MQIPT . Prenez soin d'inclure le préfixe correct (file:/) et le suffixe de fichier correct (/lib/com.ibm.mq.ipt.jar). Sur les systèmes AIX and Linux , une URL **codeBase** standard peut être file:/opt/mqipt/lib/com.ibm.mq.ipt.jar, en supposant que MQIPT est installé dans /opt/mqipt.

Les droits sont généralement définis avec trois attributs. Pour contrôler des connexions socket, leurs valeurs sont :

droit de classe

java.net.SocketPermission

nom à contrôler

Il est composé du format hostname : port, dans lequel chaque composant du nom peut être spécifié par un caractère générique. Le nom d'hôte peut être un nom de domaine ou l'adresse IP. La position la plus à gauche du nom d'hôte peut être indiquée par un astérisque (*). Par exemple, harry.company1.com serait mis en correspondance par chacune des chaînes suivantes:

- harry
- harry.company1.com
- *.company1.com
- *

- 198.51.100.123 (en supposant que ce soit l'adresse IP de harry.company1.com)

Le composant de port de ce nom peut être spécifié comme une seule adresse de port ou une plage d'adresses de port, par exemple :

1414

uniquement le port 1414

1414-

toutes les adresses de port supérieures ou égales à 1414

-1414

toutes les adresses de port inférieures ou égales à 1414

1-1414

toutes les adresses de port comprises entre 1 et 1414 inclus

action autorisée

Les actions utilisées par `java.net.SocketPermission` sont :

accepter

Autoriser l'acceptation des connexions à partir de la cible spécifiée

connect

Autoriser les connexions à la cible spécifiée

être en mode écoute

Autoriser l'application à écouter sur le ou les ports spécifiés pour les demandes de connexion

résoudre

Autoriser l'utilisation de DNS pour résoudre les noms de domaine en adresses IP

Le contrôle de Java security manager peut également être effectué via les propriétés système `java.security.manager` et `java.security.policy` Java, mais il est recommandé d'utiliser les propriétés **SecurityManager** et **SecurityManagerPolicy** pour contrôler MQIPT.

Pour inclure des informations de diagnostic dans la trace et dans les enregistrements FFST, MQIPT doit accéder à certaines propriétés système et variables d'environnement MQIPT. Vous devez toujours inclure les propriétés suivantes dans la règle de sécurité Java :

```
permission java.util.PropertyPermission "java.home", "read";
permission java.util.PropertyPermission "java.version", "read";
permission java.util.PropertyPermission "java.runtime.version", "read";
permission java.util.PropertyPermission "java.vm.info", "read";
permission java.util.PropertyPermission "java.vm.vendor", "read";
permission java.util.PropertyPermission "os.arch", "read";
permission java.util.PropertyPermission "os.name", "read";
permission java.util.PropertyPermission "os.version", "read";
permission java.lang.RuntimePermission "getenv.MQIPT_PATH";
permission java.lang.RuntimePermission "getStackTrace";
permission javax.management.MBeanServerPermission "createMBeanServer";
permission javax.management.MBeanPermission "com.ibm.mq.ipt.IPTManager#[com.ibm.mq.ipt:type=IPTManager]", "registerMBean";
permission javax.management.MBeanPermission "com.ibm.mq.ipt.IPTManager#[com.ibm.mq.ipt:type=IPTManager]", "unregisterMBean";
permission javax.management.MBeanTrustPermission "register";
```

Si vous n'incluez pas toutes ces propriétés, MQIPT ne fonctionnera pas correctement, et le diagnostic des problèmes sera affecté.

Exits de sécurité dans MQIPT

Utilisez un exit de sécurité pour contrôler l'accès à une destination cible, définie par la propriété de route **Destination**. L'exit de sécurité est appelé au point où MQIPT reçoit une demande de connexion d'un client, mais avant d'effectuer la connexion sur la destination cible.

En fonction des propriétés de connexion initiales, l'exit de sécurité décide si la connexion est autorisée à se terminer.

Lorsqu'un chemin est démarré, l'exit de sécurité est appelé afin de s'initialiser et pour se préparer à traiter une demande de connexion. Le processus d'initialisation doit être utilisé pour charger les données utilisateur et préparer ces données pour accélérer et faciliter l'accès. Cela accélère le traitement des demandes de connexion.

Chaque route peut avoir son propre exit de sécurité.

- La propriété **SecurityExit** active/désactive l'exit de sécurité défini par l'utilisateur.
- La propriété **SecurityExitName** définit le nom de classe de l'exit de sécurité défini par l'utilisateur.
- La propriété **SecurityExitPath** définit le nom du répertoire contenant le fichier de classe. Si cette propriété n'est pas définie, le système suppose que le fichier de classe se trouve dans le sous-répertoire des exits. **SecurityExitPath** peut également définir le nom d'un fichier JAR contenant l'exit de sécurité défini par l'utilisateur.
- La propriété **SecurityExitTimeout** est utilisée par MQIPT pour déterminer le temps d'attente d'une réponse de l'exit de sécurité lors de la validation d'une demande de connexion.

Pour plus d'informations sur les propriétés de l'exit de sécurité, voir [MQIPT route properties](#).

MQIPT utilise la classe `SecurityExit` pour appeler un exit de sécurité défini par l'utilisateur. Cette classe doit être étendue par l'exit de sécurité défini par l'utilisateur. La plupart de ses méthodes doivent être remplacées pour fournir la fonctionnalité requise. Un objet `SecurityExitResponse` est utilisé pour retransmettre les données à MQIPT. MQIPT exploite ces données pour déterminer si la demande de connexion doit être acceptée ou rejetée. L'objet `SecurityExitResponse` peut également contenir une nouvelle destination et une nouvelle adresse de port de destination, utilisées pour remplacer la route définie par les propriétés de l'exit de sécurité.

Trois exemples d'exit de sécurité sont fournis pour vous montrer comment implémenter un exit de sécurité.

- `SampleSecurityExit` indique comment contrôler l'accès à un gestionnaire de files d'attente d'IBM MQ, en fonction du nom du canal IBM MQ. Il autorise uniquement une connexion avec un nom de canal commençant par la chaîne "MQIPT." Pour plus d'informations, voir [Utilisation d'un exit de sécurité](#).
- `SampleRoutingExit` permet le routage dynamique de demandes de connexion client à un pool de serveurs IBM MQ définis. Chaque serveur héberge un gestionnaire de files d'attente du même nom avec les mêmes attributs. L'exemple comprend un fichier de configuration qui contient la liste des noms de serveur. Pour plus d'informations, voir [Routage des demandes de connexion client vers les serveurs de gestionnaire de files d'attente IBM MQ à l'aide des exits de sécurité](#).
- `SampleOneRouteExit` permet un routage dynamique vers un gestionnaire de files d'attente IBM MQ dérivé du nom de canal IBM MQ utilisé dans la demande de connexion. L'exemple comprend un fichier de configuration qui contient une mappe associant les noms des gestionnaires de files d'attente aux noms de serveur. Pour plus d'informations, voir [Routage dynamique des demandes de connexion client](#).

Remarque : MQIPT s'exécute dans une machine virtuelle Java unique. Un exit de sécurité défini par l'utilisateur risque de compromettre le fonctionnement normal de MQIPT de l'une des façons suivantes :

- Impact sur les ressources système
- Génération de goulots d'étranglement
- Dégradation des performances

Vous devez tester les effets de votre exit de sécurité de manière extensive avant de l'implémenter dans un environnement de production.

La classe `com.ibm.mq.ipt.exit.SecurityExit` dans MQIPT

Cette classe et ses méthodes publiques doivent être étendues par l'exit de sécurité défini par l'utilisateur pour obtenir l'accès à certaines données communes et autoriser certaines initialisations de MQIPT.

Avant l'appel de chaque méthode par MQIPT, certaines propriétés sont mises à disposition de la méthode. Leurs valeurs peuvent être extraites à l'aide des méthodes `get` appropriées définies dans cette classe.

Méthodes

public int init(IPTTrace)

Les propriétés suivantes sont disponibles :

- port du programme d'écoute
- destination
- port de destination
- version

La méthode `init` est appelée par MQIPT lorsqu'une route est démarrée. Dans le retour de cette méthode, l'exit de sécurité doit être prêt à valider une demande de connexion. Les codes retour valides sont `ExitRc.OK` ou `ExitRc.INIT_ERROR`.

public int refresh(IPTTrace)

Les propriétés suivantes sont disponibles :

- port du programme d'écoute
- destination
- port de destination

La méthode `refresh` est appelée par MQIPT lorsque la configuration MQIPT est actualisée. Cette action est généralement effectuée lorsqu'une propriété a été modifiée dans le fichier de configuration. MQIPT recharge toutes les propriétés du fichier de configuration pour déterminer quelles propriétés ont été modifiées et si une route doit être redémarrée.

Cette méthode doit effectuer un rechargement des données externes qu'elle utilise, c'est-à-dire des données chargées par la méthode `init`. Les codes retour valides sont `ExitRc.OK` ou `ExitRc.REFRESH_ERROR`.

public void close(IPTTrace)

Les propriétés suivantes sont disponibles :

- port du programme d'écoute
- destination
- port de destination

La méthode `close` est appelée par MQIPT lorsqu'elle est en cours d'arrêt. Cette méthode doit libérer toutes les ressources système que l'exit a acquises lors de son fonctionnement. MQIPT attend que cette méthode soit terminée avant de s'arrêter.

Cette méthode est également appelée si un exit de sécurité a été précédemment activé, mais qu'il a été désactivé dans le fichier de configuration.

public SecurityExitResponse validate(IPTTrace)

Les propriétés suivantes sont disponibles :

- port du programme d'écoute
- destination
- port de destination
- délai d'attente
- adresse IP du client
- adresse port du client
- nom de canal
- nom de gestionnaire de files d'attente

La méthode `validate` est appelée par MQIPT lorsqu'elle reçoit une demande de connexion à valider. Le nom de canal et le nom de gestionnaire de files d'attente ne seront pas disponibles si la propriété **SSLProxyMode** a été activée, car cette fonction n'est utilisée que pour tunneller les données TLS et, par conséquent, les données généralement obtenues à partir du flux de données initial sont illisibles.

L'exit de sécurité doit renvoyer un objet `SecurityExitResponse` contenant les informations suivantes:

- code anomalie (doit être défini)
- nouvelle adresse de destination (facultative)
- nouvelle adresse du port d'écoute de destination (facultative)
- message (facultatif)

Le code anomalie détermine si la connexion est acceptée ou rejetée par MQIPT. Les zones `newDestination` et `newDestinationPort` peuvent éventuellement être définies pour définir un nouveau gestionnaire de files d'attente cible. Si vous ne définissez pas ces propriétés, les propriétés de route **Destination** et **DestinationPort** définies dans le fichier de configuration sont utilisées. Tout message renvoyé est ajouté à l'entrée du fichier journal de connexion.

Les méthodes suivantes sont prises en charge pour l'obtention des valeurs des propriétés de configuration MQIPT :

public int getListenerPort()

extrait le port d'écoute de route-tel que défini par la propriété **ListenerPort**

public String getDestination()

extrait l'adresse de destination, telle que définie par la propriété **Destination**

public int getDestinationPort()

extrait l'adresse du port d'écoute de destination, telle que définie par la propriété **DestinationPort**

public String getClientIPAddress()

Extrait l'adresse IP du client à l'origine de la demande de connexion.

public int getClientPortAddress()

Extrait l'adresse de port utilisée par le client à l'origine de la demande de connexion.

public int getTimeout()

Extrait la valeur du délai d'attente. MQIPT attend que l'exit de sécurité valide une demande, comme défini par la propriété **SecurityExitTimeout**

public int getConnThreadID()

Extrait l'ID d'unité d'exécution de connexion qui gère la demande de connexion, ce qui est pratique pour le débogage

public String getChannelName()

Extrait le nom du canal IBM MQ utilisé dans la demande de connexion.

public String getQMName()

Extrait le nom de gestionnaire de files d'attente IBM MQ utilisé dans la demande de connexion.

public boolean getTimedout()

Utilisé par l'exit de sécurité pour déterminer si le délai d'attente a expiré.

La classe `com.ibm.mq.ipt.exit.SecurityExitResponse` dans MQIPT

Cette classe est utilisée pour retransmettre une réponse à MQIPT à partir d'un exit de sécurité défini par l'utilisateur. Elle est utilisée pour déterminer si la demande de connexion doit être acceptée ou rejetée.

Les objets de ce type sont créés uniquement dans la méthode de validation (voir «[La classe `com.ibm.mq.ipt.exit.SecurityExit` dans MQIPT](#)», à la page 1101). Il existe des constructeurs tous faits pour créer ces objets ainsi que des méthodes pour chaque propriété. Pour plus d'informations, voir les exemples d'exit de sécurité.

La création d'un objet `SecurityExitResponse` par défaut rejette la demande de connexion.

Constructeurs

- **public SecurityExitResponse (String dest, int destPort, int rc, String msg)**

où :

- `dest` est la nouvelle destination cible

- `destPort` est la nouvelle adresse du port de destination
- `rc` est le code anomalie
- `msg` est un message qui sera ajouté à la l'entrée du journal de connexion
- **public SecurityExitResponse (String dest, int destPort, int rc)**
- **public SecurityExitResponse (int rc, String msg)**
- **public SecurityExitResponse (int rc)**

Méthodes

public void setDestination(String dest)

définit une nouvelle adresse de destination pour la demande de connexion.

public void setDestinationPort(int port) throws IPTException

définit une nouvelle adresse de port d'écoute de destination pour la demande de connexion, et émet une IPTException pour une adresse de port non valide.

public void setMessage(String msg)

ajoute un message à l'enregistrement de journal de connexion.

public void setReasonCode(int rc)

définit le code raison de la demande de connexion.

Codes retour de l'exit de sécurité dans MQIPT

Codes retour reconnus par MQIPT lors de l'appel d'un exit de sécurité dans un certain nombre de situations différentes.

Les codes retour suivants sont reconnus par MQIPT lors de l'appel d'un exit de sécurité dans les situations suivantes :

Code retour	Description	init	Validat e	Actualis er
ExitRc.OK	L'exécution de la demande a abouti.	oui	oui	oui
ExitRc.INIT_ERROR	Echec de la demande d'initialisation, la route sera désactivée.	oui		
ExitRc.REFRESH_ERROR	Echec de la demande d'actualisation.			oui
ExitRc.NOT_AUTHORIZED	Le processus de validation a échoué, demande de connexion rejetée.		oui	
ExitRc.DISABLE_SSL	La demande de validation a abouti, la connexion à la cible n'utilisera pas SSL ou TLS.		oui	

Contrôle de numéro de port dans MQIPT

Lors de l'utilisation de MQIPT, il est possible de limiter la plage des numéros de port locaux utilisés dans une connexion sortante.

Définissez la propriété **OutgoingPort** sur la route afin de spécifier le numéro de port local initial, et définissez **MaxConnectionThreads** pour spécifier le nombre de ports à utiliser. Par exemple, si vous définissez **OutgoingPort** sur 1600 et **MaxConnectionThreads** sur 20, la plage des numéros de port locaux utilisés pour cette route est comprise entre 1600 et 1619.

Il incombe à l'administrateur de MQIPT de vérifier qu'il n'y a aucun conflit de numéros de port entre les routes.

Si la propriété **OutgoingPort** n'est pas définie, une valeur par défaut de 0 indique qu'un numéro de port attribué par le système est utilisé pour chaque connexion.

Lorsque vous utilisez HTTP, le nombre de ports sortant est deux fois supérieur au nombre de ports utilisés avec d'autres protocoles. Dans l'exemple précédent, si la route utilisait HTTP, la plage des numéros est comprise entre 1600 et 1639.

Pour plus d'informations, voir [Allocation de numéros de port](#).

Systemes multihébergés

Lorsque vous utilisez un système multihébergé, vous pouvez spécifier l'adresse IP à laquelle une connexion sortante se connectera, à l'aide de la propriété **LocalAddress**. Les noms d'hôte ne sont pas pris en charge dans cette propriété.

Chiffrement des mots de passe stockés dans MQIPT

La configuration MQIPT peut inclure des mots de passe permettant d'accéder à diverses ressources, ainsi que le mot de passe permettant d'accéder à MQIPT à l'aide du port de commande. Tous ces mots de passe doivent être protégés en étant chiffrés.

Pourquoi et quand exécuter cette tâche

Tous les mots de passe stockés devant être utilisés par MQIPT doivent être protégés en chiffrant le mot de passe à l'aide de la commande **mqiptPW**. Les mots de passe chiffrés sont stockés en tant que valeurs de propriété dans le fichier de configuration **mqipt.conf**. MQIPT permet de faire la distinction entre les mots de passe chiffrés, les mots de passe en texte clair et les noms de fichier dans les valeurs de propriété. Vous devez chiffrer tous les mots de passe stockés pour être utilisés par MQIPT de cette manière, car il s'agit de la méthode de protection la plus sécurisée.

Pour améliorer la protection des mots de passe de fichier de clés, chiffrez à nouveau les mots de passe de fichier de clés précédemment chiffrés à l'aide de la méthode de protection la plus récente.

Remarque : La propriété **SSLCommandPortKeyRingPW** du fichier de configuration **mqipt.conf** et la propriété **SSLClientCAKeyRingPW** du fichier de propriétés **mqiptAdmin** ne peuvent pas faire référence à des fichiers de mot de passe. Les valeurs de ces propriétés doivent être définies sur la chaîne de mot de passe chiffré générée par la commande **mqiptPW**.

Si un texte en clair ou un mot de passe faiblement protégé est présent dans la configuration MQIPT, un message d'avertissement est émis lorsque MQIPT démarre ou lorsqu'une route démarre.

Utilisez cette procédure pour chiffrer un mot de passe à stocker pour une utilisation par MQIPT à l'aide de la méthode de protection la plus récente.

Procédure

1. Facultatif : Créez un fichier contenant la clé de chiffrement de mot de passe, si vous n'en avez pas déjà une.

MQIPT utilise une clé de chiffrement pour chiffrer les mots de passe. Vous pouvez spécifier votre propre clé de chiffrement dans un fichier. Le fichier doit contenir au moins un caractère et une seule ligne de texte.

La même clé de chiffrement de mot de passe est utilisée pour chiffrer et déchiffrer tous les mots de passe stockés pour une instance de MQIPT. Par conséquent, vous n'avez besoin que d'un seul fichier de clé de chiffrement de mot de passe pour chaque installation MQIPT.

Vous pouvez utiliser une autre clé de chiffrement de mot de passe pour chiffrer les mots de passe stockés dans le fichier de propriétés **mqiptAdmin** que la clé de chiffrement utilisée pour chiffrer les mots de passe dans la configuration MQIPT.

Si vous prévoyez d'exécuter MQIPT en tant que service démarré automatiquement, vous devez créer le fichier de clé de chiffrement de mot de passe avec le nom par défaut **mqipt_cred.key** et le placer dans le répertoire de base MQIPT.

Vous n'avez pas besoin de spécifier une clé de chiffrement de mot de passe, mais cette opération est plus sécurisée. Si vous ne spécifiez pas votre propre clé de chiffrement, la clé de chiffrement par défaut est utilisée.

Remarque : Vous devez vous assurer que les droits d'accès aux fichiers appropriés sont définis sur le fichier de clé de chiffrement de mot de passe pour empêcher les utilisateurs non autorisés de lire la clé de chiffrement. Seul l'utilisateur qui exécute la commande **mqiPTPW** et l'utilisateur sous lequel MQIPT s'exécute doivent être autorisés à lire la clé de chiffrement de mot de passe.

2. Chiffrez le mot de passe à l'aide de la commande **mqiPTPW**.

La syntaxe de la commande **mqiPTPW** est décrite dans [mqiPTPW \(chiffrement du mot de passe stocké\)](#).

Si vous avez créé un fichier de clé de chiffrement de mot de passe à l'étape «1», à la page 1105, spécifiez le nom de fichier à l'aide du paramètre **-sf** sur **mqiPTPW**. Par exemple, la commande suivante peut être émise pour chiffrer un mot de passe à l'aide de la clé de chiffrement dans le fichier spécifié par le paramètre **-sf** :

```
mqiPTPW -sf /opt/mqiPT/mqiPT_password.key
```

3. Entrez le mot de passe à chiffrer lorsque vous y êtes invité.

Le mot de passe chiffré sera généré par **mqiPTPW**.

4. Copiez le mot de passe chiffré dans la propriété appropriée du fichier de configuration `mqiPT.conf` ou dans le fichier de propriétés **mqiPTAdmin**.

Par exemple, la ligne suivante indique un mot de passe chiffré pour le mot de passe d'accès MQIPT :

```
AccessPW=<mqiPTPW>1!QL+2Jvj/tigKK1D7Nz80qw==!AMDBef0UrmPf5i10uqV5MA==
```

5. Démarrez MQIPT. Si vous avez créé un fichier de clés de chiffrement de mot de passe à l'étape «1», à la page 1105 avec un nom autre que le nom par défaut, indiquez le nom du fichier de clés de chiffrement lors du démarrage de MQIPT.

Vous pouvez spécifier le nom du fichier de clés de chiffrement de mot de passe à l'aide du paramètre **-sf** lors du démarrage de MQIPT. Par exemple, exécutez la commande suivante pour démarrer MQIPT à l'aide de la clé de chiffrement dans le fichier spécifié par le paramètre **-sf** :

```
mqiPT /opt/mqiPT -sf /opt/mqiPT/mqiPT_password.key
```

Pour plus d'informations sur les autres méthodes permettant de spécifier le nom du fichier de clé de chiffrement de mot de passe lors du démarrage de MQIPT, voir [Spécification de la clé de chiffrement de mot de passe](#).

Vous pouvez spécifier le nom du fichier de clé de chiffrement de mot de passe pour la commande **mqiPTAdmin** à l'aide de la propriété **PasswordProtectionKeyFile** dans le fichier de propriétés **mqiPTAdmin**.

Autres considérations de sécurité pour MQIPT

MQIPT propose des fonctions supplémentaires qui permettent de concevoir une solution sécurisée.

- S'il existe plusieurs clients dans un réseau interne, tous tentant de créer des connexions sortantes, ils peuvent tous passer par un MQIPT situé à l'intérieur du pare-feu. L'administrateur du pare-feu doit ensuite accorder un accès externe uniquement à l'ordinateur MQIPT.
- MQIPT peut se connecter uniquement aux gestionnaires de files d'attente pour lesquels il a été configuré explicitement dans son fichier de configuration, sauf si MQIPT agit en tant que proxy SOCKS ou s'il utilise un exit de sécurité.
- MQIPT vérifie que les messages qu'il reçoit et transmet sont valides, et conformes au protocole IBM MQ. Cela permet d'éviter les attaques de sécurité exploitant MQIPT en dehors du protocole IBM MQ. Si MQIPT agit en tant que proxy SSL/TLS, lorsque toutes les données et tous les protocoles IBM MQ ont été chiffrés, MQIPT ne peut garantir que l'établissement de liaison SSL/TLS initial. Dans ce cas, utilisez [Java security manager](#).
- MQIPT permet à des exits de canal d'exécuter leurs propres protocoles de sécurité de bout en bout.

- Vous pouvez limiter le nombre total de connexions entrantes en définissant la propriété `MaxConnectionThreads`. Elle permet de protéger un gestionnaire de files d'attente interne vulnérable contre les attaques DoS (déni de service).

Fichier de configuration

Vous devez protéger le fichier de configuration de MQIPT (`mqipt.conf`), contre les tentatives de lecture par des utilisateurs non autorisés, car il est susceptible de contenir des informations sensibles, telles que le mot de passe **AccessPW** qui contrôle les droits d'administration à distance de MQIPT. Protégez tous les mots de passe spécifiés dans le fichier de configuration en suivant la procédure décrite dans [«Chiffrement des mots de passe stockés dans MQIPT»](#), à la page 1105. Vérifiez également que `mqipt.conf` est protégé contre les modifications non autorisées. Définissez les droits d'accès aux fichiers du système d'exploitation pour `mqipt.conf` de sorte que seul le compte utilisateur qui exécute MQIPT puisse lire ou mettre à jour le fichier.

Port de commande

Les ports de commande MQIPT acceptent les commandes d'administration émises sur le réseau vers une instance distante de MQIPT par la commande **mqiptAdmin**.

MQIPT peut être configuré avec un port de commande non sécurisé et un port de commande sécurisé avec TLS. Les connexions au port de commande non sécurisé ne sont pas chiffrées.

Remarque : Les données envoyées sur le réseau au port de commande non sécurisé, y compris le mot de passe d'accès MQIPT, peuvent être visibles par d'autres utilisateurs sur le réseau.

Vous devez déterminer si vous devez activer un port de commande et évaluer les risques liés à l'autorisation de l'administration à distance de MQIPT, avant d'activer le port de commande non sécurisé ou TLS. La commande **mqiptAdmin** peut administrer des instances locales de MQIPT qui s'exécutent sous le même nom d'utilisateur que la commande **mqiptAdmin** sans utiliser de port de commande. Par conséquent, il se peut que vous n'ayez pas besoin d'activer un port de commande pour administrer les instances locales de MQIPT.

Si le port de commande non sécurisé ou TLS est activé, vous devez empêcher tout accès non autorisé au port de commande. Par exemple, vous devez tenir compte des points suivants lors de la sécurisation de l'accès au port de commande:

- Utilisez un pare-feu pour restreindre l'ensemble des ordinateurs pouvant se connecter au port de commande MQIPT.
- Activez l'authentification sur les ports de commande à l'aide des propriétés **AccessPW** et **RemoteCommandAuthentication**. Pour plus d'informations sur l'activation de l'authentification par port de commande, voir [Authentification par port de commande](#).
- Désactivez l'arrêt à distance à l'aide de la propriété **RemoteShutdown**.
- Envisagez d'utiliser les propriétés **CommandPortListenerAddress** et **SSLCommandPortListenerAddress** pour configurer les ports de commande afin qu'ils soient à l'écoute sur une interface réseau spécifique.

Pour plus d'informations sur l'utilisation de la commande **mqiptAdmin** pour administrer MQIPT, voir [Administration MQIPT à l'aide de la ligne de commande](#).

Journaux de connexion dans MQIPT

MQIPT fournit une fonction de journal de connexion qui contient les listes de toutes les tentatives de connexion ayant abouti ou échoué.

Une entrée est écrite dans le journal de connexion pour chaque connexion reçue ou établie par une route MQIPT et pour chaque commande d'administration reçue par MQIPT. Le journal de connexion est contrôlé à l'aide des propriétés **ConnectionLog** et **MaxLogFileSize**. Pour plus d'informations, voir [Propriétés globalesMQIPT](#).

A chaque démarrage de MQIPT, une nouvelle connexion journal est créée. Pour des raisons d'identification, le nom de fichier inclut l'horodatage en cours, par exemple :

```
mqiptYYYYMMDDHHmmSS.log
```

Où

- YYYY est l'année
- MM indique le mois.
- DD est le jour
- HH indique les heures.
- mm indique les minutes.
- SS indique les secondes.

Lorsqu'un journal des connexions atteint la taille maximale définie par la propriété **MaxLogFileSize**, un fichier de sauvegarde, `mqipt001.log`, est créé. Deux fichiers de sauvegarde au maximum sont conservés (`mqipt001.log` et `mqipt002.log`).

Une entrée dans le journal de connexion représente chaque partie d'une demande de connexion. Une demande de connexion reçue par MQIPT; ainsi que la nouvelle connexion établie par MQIPT à l'adresse de destination s'affiche sous la forme de deux entrées de journal, puis de deux autres entrées lorsque chaque connexion est arrêtée.

Voici le journal des connexions en cas de demande de connexion réussie :

```
Wed May 15 13:13:51 BST 2013 conn accept 127.0.0.1(3842) 127.0.0.1(5000) OK 5000-0
Wed May 15 13:13:51 BST 2013 conn conn 127.0.0.1(3843) localhost(3500) OK 5000-0
Wed May 15 13:13:52 BST 2013 conn close 127.0.0.1(3842) 127.0.0.1(5000) OK 5000-0
Wed May 15 13:13:52 BST 2013 conn close 127.0.0.1(3843) localhost(3500) OK 5000-0
```

Voici le journal des connexions en cas d'échec d'une demande de connexion :

```
Wed May 15 14:56:40 BST 2013 conn accept 127.0.0.1(4138) 127.0.0.1(7000) OK 7000-0
Wed May 15 14:56:40 BST 2013 conn close 127.0.0.1(4138) 127.0.0.1(7000) ERROR 7000-0
Unrecognized SSL handshake request '54'
```

Entrées du journal de connexion

Chaque entrée du journal de connexion contient les informations suivantes:

- Heure de création de l'entrée.
- Type d'entrée. Les valeurs admises sont les suivantes :

admin

Commande d'administration

CONN

Connexion de route

- Événement qui s'est produit. Les valeurs admises sont les suivantes :

accepter

Demande de connexion reçue

fermer

Connexion fermée

CONN

Demande de connexion à la destination de routage

dspipt

Commande MQIPT d'affichage reçue

NoData

Aucune donnée reçue de l'appelant

commande PING

Demande ping reçue

statut

Commande d'affichage de l'état reçue

refr

Commande d'actualisation reçue

arrêt

Commande d'arrêt reçue

- Adresse du réseau source et numéro de port. La valeur LOCAL est affichée pour les commandes d'administration émises localement sans utiliser le port de commande.
- Adresse réseau de destination et numéro de port. Cette option n'est pas affichée pour les commandes d'administration émises localement sans utiliser le port de commande.
- Code achèvement. La valeur peut être OK ou ERROR.
- Identificateur de l'unité d'exécution MQIPT .
- Message d'erreur facultatif.

Configuration de IBM MQ Internet Pass-Thru à l'aide de conteneurs

Vous pouvez exécuter IBM MQ Internet Pass-Thru (MQIPT) dans un conteneur. L'image de base utilisée par l'image de conteneur doit utiliser un système d'exploitation Linux qui est pris en charge.

Procédure

- Un exemple d'image MQIPT Docker est disponible dans le référentiel GitHub [mq-container](#). Pour générer et exécuter le conteneur, suivez les instructions de la rubrique [IBM MQ Internet Pass-Thru sur Docker](#).

Que faire ensuite

Vous pouvez afficher les conteneurs en cours d'exécution à l'aide de la commande **docker ps** . Pour afficher la sortie de la console de MQIPT s'exécutant dans un conteneur Docker , utilisez la commande **docker logs \${CONTAINER_ID}** .

Configuration des files d'attente de flux

La fonction de files d'attente de diffusion en flux vous permet de disposer d'une copie en double de chaque message inséré dans une file d'attente, distribuée dans une seconde file d'attente. La configuration des files d'attente de diffusion en flux est effectuée file par file d'attente.

Les files d'attente locales et modèles possèdent deux nouveaux attributs liés aux files d'attente de diffusion en flux:

STREAMQ

Il s'agit du nom de la file d'attente à laquelle les messages diffusés en continu doivent être distribués. Vous devez définir l'attribut **STREAMQ** sur le nom d'une autre file d'attente.

Il existe des restrictions sur les files d'attente qui peuvent être configurées pour transmettre des messages à d'autres files d'attente, et des restrictions sur les files d'attente qui peuvent être définies comme destination pour les messages diffusés en flux. Pour plus d'informations sur les restrictions de diffusion en flux de messages, voir [Restrictions de file d'attente de flux](#).

STRMQOS

Il s'agit de la qualité de service à utiliser lors de la distribution de messages diffusés en continu.

Vous pouvez définir l'attribut **STRMQOS** sur l'une des deux valeurs suivantes:

BESTEF

Meilleur effort, qui est la valeur par défaut.

Le gestionnaire de files d'attente tente de distribuer une copie de chaque message à la file d'attente spécifiée dans l'attribut **STREAMQ**. S'il y a un problème lors de la distribution du message diffusé en continu, cela n'affecte pas la distribution du message d'origine.

MUSTDUP

Le gestionnaire de files d'attente tente de distribuer une copie de chaque message à la file d'attente de diffusion en flux.

En cas de problème lors de la distribution du message diffusé en flux, le message d'origine n'est pas distribué dans sa file d'attente et l'application reçoit MQCC_FAILED avec un code anomalie approprié.

Pour plus de détails, voir les commandes [ALTER queues](#), [DEFINE queues](#) et [DISPLAY QUEUE MQSC](#), ainsi que les commandes [Change, Copy, and Create Queue](#), [Inquire Queue](#) et [Inquire Queue \(Response\) PCF](#).

Si plusieurs copies de chaque message sont requises, vous pouvez configurer l'attribut **STREAMQ** pour qu'il fasse référence au nom d'une file d'attente alias IBM MQ dont la cible fait référence à une rubrique IBM MQ. Lorsqu'un message est inséré dans la file d'attente d'origine, une copie du message est publiée dans la rubrique nommée.

Vous devez vous assurer que vous disposez d'abonnements API ou administrés à l'objet de rubrique lorsque chaque abonnement reçoit une copie du message. Le message distribué aux abonnés suit les mêmes règles que les autres messages de publication / abonnement. Par exemple, chaque message possède un nouvel identificateur de message et les zones de contexte du MQMD sont différentes de celles du message d'origine. Pour plus d'informations sur les similitudes et les différences entre les messages d'origine et les messages diffusés en continu, voir [Messages diffusés en continu](#).

Exemples

Exemple de meilleur effort

Dans l'exemple suivant, une file d'attente locale ORDERS.QUEUE est modifiée pour insérer des messages en flux dans une seconde file d'attente ANALYTICS.QUEUE. La qualité de service BESTEF est utilisée pour s'assurer qu'en cas de problème lors de l'insertion du message transmis dans ANALYTICS.QUEUE, par exemple, si ANALYTICS.QUEUE est pleine, le message d'origine peut toujours être inséré dans ORDERS.QUEUE.

Ce type de configuration peut être utilisé pour effectuer des analyses sur les commandes reçues, en analysant les messages transmis, tandis que les messages d'origine sont placés dans la file d'attente de commandes et sont traités. L'un des avantages de la fonction de file d'attente de diffusion en flux est que vous pouvez laisser les messages diffusés en flux sur ANALYTICS.QUEUE en attente de traitement, sans affecter les commandes réelles satisfaites par l'entreprise.

```
DEFINE QLOCAL (ANALYTICS.QUEUE)
```

```
ALTER QLOCAL (ORDERS.QUEUE) STRMQOS (BESTEF) STREAMQ (ANALYTICS.QUEUE)
```

Remarque : Dans l'exemple, **STRMQOS** a été défini sur BESTEF, bien que vous puissiez laisser cet attribut hors de la commande **ALTER**, car BESTEF est la qualité de service par défaut.

Exemple en double obligatoire

Dans cet exemple, une file d'attente locale PAYMENTS.QUEUE est modifiée pour insérer des copies en flux de chaque message dans une autre file d'attente locale AUDIT.QUEUE. Il est important que chaque message inséré dans la file d'attente de paiement soit transmis à la file d'attente d'audit, de sorte que la qualité de service MUSTDUP soit utilisée.

S'il y a un problème lors de la distribution du message diffusé dans sa file d'attente, le message d'origine n'est pas non plus distribué et l'application reçoit un code d'achèvement et de raison approprié. L'application doit relancer l'insertion de la même manière que si une seule file d'attente était impliquée.

```
DEFINE QLOCAL (AUDIT.QUEUE)
```

```
ALTER QLOCAL (PAYMENTS.QUEUE) STRMQOS (MUSTDUP) STREAMQ (AUDIT.QUEUE)
```

Remarques :

1. Il n'est pas nécessaire que la file d'attente de diffusion en flux existe lors de la modification de la file d'attente d'origine. Cependant, il est important de noter que comme la qualité de service utilisée est MUSTDUP, les tentatives d'insertion de messages dans la file d'attente d'origine échouent tant que vous n'avez pas défini la file d'attente de diffusion en flux.
2. Lors de l'utilisation d'un alias de file d'attente avec une cible d'un objet de rubrique, s'il n'y a pas d'abonnés, la distribution du message diffusé est toujours considérée comme réussie et le message d'origine est distribué à sa file d'attente.
3. Si un message diffusé en flux ne peut pas être distribué dans sa file d'attente, le gestionnaire de files d'attente ne tente pas de le distribuer dans la file d'attente de rebut. Toutefois, si un message diffusé en flux est envoyé à une file d'attente éloignée, lorsqu'il est acheminé via un canal vers un autre gestionnaire de files d'attente, il peut être distribué à une file d'attente de rebut en suivant les règles de rebut existantes.

Configuration de la file d'attente en flux

Aucune configuration supplémentaire n'a besoin d'être effectuée sur la file d'attente de diffusion en flux. Il reçoit des messages de toutes les files d'attente qui le nomment comme file d'attente de diffusion en flux. Toutefois, il peut être judicieux de prendre en compte les valeurs d'attribut configurées dans la file d'attente de diffusion en flux.

Par exemple, si la longueur maximale de la file d'attente d'origine est de 100 000 et que la longueur maximale de la file d'attente de diffusion en flux est de 5000, les messages diffusés en flux risquent d'être perdus si STRMQOS est défini sur BESTEF, ou d'échouer au moment de l'insertion si STRMQOS est défini sur MUSTDUP, avec une erreur MQRC_Q_FULL, même si la file d'attente d'origine contient beaucoup d'espace.

Tenez compte des attributs de la file d'attente de diffusion en flux qui doivent être modifiés pour avoir des valeurs appropriées, en fonction de la manière dont la file d'attente d'origine est configurée.

Concepts associés

[Files d'attente de flux](#)

Remarques

:NONE.

Le présent document peut contenir des informations ou des références concernant certains produits, logiciels ou services IBM non annoncés dans ce pays. Consultez votre représentant IBM local pour obtenir des informations sur les produits et services actuellement disponibles dans votre région. Toute référence à un produit, logiciel ou service IBM n'implique pas que seul ce produit, logiciel ou service IBM puisse être utilisé. Tout produit, programme ou service fonctionnellement équivalent qui ne porte pas atteinte à un droit de propriété intellectuelle IBM peut être utilisé à la place. Il est de la responsabilité de l'utilisateur d'évaluer et de vérifier lui-même les installations et applications réalisées avec des produits, logiciels ou services non expressément référencés par IBM.

IBM peut détenir des brevets ou des demandes de brevet couvrant les produits mentionnés dans le présent document. La remise de ce document ne vous donne aucun droit de licence sur ces brevets ou demandes de brevet. Si vous désirez recevoir des informations concernant l'acquisition de licences, veuillez en faire la demande par écrit à l'adresse suivante :

IBM Director of Licensing
IBM Corporation
North Castle Drive
Armonk, NY 10504-1785
U.S.A.

Pour obtenir des informations sur les licences relatives aux informations sur deux octets (DBCS), contactez le service de la propriété intellectuelle IBM de votre pays ou envoyez vos demandes de renseignements, par écrit, à :

Intellectual Property Licensing
Legal and Intellectual Property Law
IBM Japan, Ltd.
19-21, Nihonbashi-Hakozakicho, Chuo-ku
Tokyo 103-8510, Japan

Le paragraphe suivant ne s'applique ni au Royaume-Uni, ni dans aucun pays dans lequel il serait contraire aux lois locales. LE PRESENT DOCUMENT EST LIVRE "EN L'ETAT" SANS AUCUNE GARANTIE EXPLICITE OU IMPLICITE. IBM DECLINE NOTAMMENT TOUTE RESPONSABILITE RELATIVE A CES INFORMATIONS EN CAS DE CONTREFACON AINSI QU'EN CAS DE DEFAUT D'APTITUDE A L'EXECUTION D'UN TRAVAIL DONNE. Certaines juridictions n'autorisent pas l'exclusion des garanties implicites, auquel cas l'exclusion ci-dessus ne vous sera pas applicable.

Le présent document peut contenir des inexactitudes ou des coquilles. Ce document est mis à jour périodiquement. Chaque nouvelle édition inclut les mises à jour. IBM peut, à tout moment et sans préavis, modifier les produits et/ou programmes décrits dans ce document.

Les références à des sites Web non IBM sont fournies uniquement à titre d'information et n'impliquent en aucune façon une adhésion de ces sites Web. Les éléments figurant sur ces sites Web ne font pas partie des éléments du présent produit IBM et l'utilisation de ces sites relève de votre seule responsabilité.

IBM pourra utiliser ou diffuser, de toute manière qu'elle jugera appropriée et sans aucune obligation de sa part, tout ou partie des informations qui lui seront fournies.

Les licenciés souhaitant obtenir des informations permettant : (i) l'échange des données entre des logiciels créés de façon indépendante et d'autres logiciels (dont celui-ci), et (ii) l'utilisation mutuelle des données ainsi échangées, doivent adresser leur demande à :

IBM Corporation
Coordinateur d'interopérabilité logicielle, département 49XA
3605 Autoroute 52 N

Rochester, MN 55901
U.S.A.

Ces informations peuvent être soumises à des conditions particulières, prévoyant notamment le paiement d'une redevance.

Le logiciel sous licence décrit dans le présent document et tous les éléments sous disponibles s'y rapportant sont fournis par IBM conformément aux dispositions du Contrat sur les produits et services IBM, aux Conditions Internationales d'Utilisation de Logiciels IBM ou de tout autre accord équivalent.

Les données de performance indiquées dans ce document ont été déterminées dans un environnement contrôlé. Par conséquent, les résultats peuvent varier de manière significative selon l'environnement d'exploitation utilisé. Certaines mesures évaluées sur des systèmes en cours de développement ne sont pas garanties sur tous les systèmes disponibles. En outre, elles peuvent résulter d'extrapolations. Les résultats peuvent donc varier. Il incombe aux utilisateurs de ce document de vérifier si ces données sont applicables à leur environnement d'exploitation.

Les informations concernant des produits non IBM ont été obtenues auprès des fournisseurs de ces produits, par l'intermédiaire d'annonces publiques ou via d'autres sources disponibles. IBM n'a pas testé ces produits et ne peut confirmer l'exactitude de leurs performances ni leur compatibilité. Elle ne peut recevoir aucune réclamation concernant des produits non IBM. Toute question concernant les performances de produits non IBM doit être adressée aux fournisseurs de ces produits.

Toute instruction relative aux intentions d'IBM pour ses opérations à venir est susceptible d'être modifiée ou annulée sans préavis, et doit être considérée uniquement comme un objectif.

Le présent document peut contenir des exemples de données et de rapports utilisés couramment dans l'environnement professionnel. Ces exemples mentionnent des noms fictifs de personnes, de sociétés, de marques ou de produits à des fins illustratives ou explicatives uniquement. Toute ressemblance avec des noms de personnes, de sociétés ou des données réelles serait purement fortuite.

Licence sur les droits d'auteur :

Le présent logiciel contient des exemples de programmes d'application en langage source destinés à illustrer les techniques de programmation sur différentes plateformes d'exploitation. Vous avez le droit de copier, de modifier et de distribuer ces exemples de programmes sous quelque forme que ce soit et sans paiement d'aucune redevance à IBM, à des fins de développement, d'utilisation, de vente ou de distribution de programmes d'application conformes aux interfaces de programmation des plateformes pour lesquels ils ont été écrits ou aux interfaces de programmation IBM. Ces exemples de programmes n'ont pas été rigoureusement testés dans toutes les conditions. Par conséquent, IBM ne peut garantir expressément ou implicitement la fiabilité, la maintenabilité ou le fonctionnement de ces programmes.

Si vous visualisez ces informations en ligne, il se peut que les photographies et illustrations en couleur n'apparaissent pas à l'écran.

Documentation sur l'interface de programmation

Les informations d'interface de programmation, si elles sont fournies, sont destinées à vous aider à créer un logiciel d'application à utiliser avec ce programme.

Ce manuel contient des informations sur les interfaces de programmation prévues qui permettent au client d'écrire des programmes pour obtenir les services d'IBM MQ.

Toutefois, lesdites informations peuvent également contenir des données de diagnostic, de modification et d'optimisation. Ces données vous permettent de déboguer votre application.

Important : N'utilisez pas ces informations de diagnostic, de modification et d'optimisation en tant qu'interface de programmation car elles sont susceptibles d'être modifiées.

Marques

IBM, le logo IBM, ibm.com, sont des marques d'IBM Corporation dans de nombreux pays. La liste actualisée de toutes les marques d'IBM est disponible sur la page Web "Copyright and trademark".

information"www.ibm.com/legal/copytrade.shtml. Les autres noms de produits et de services peuvent être des marques d'IBM ou d'autres sociétés.

Microsoft et Windows sont des marques de Microsoft Corporation aux Etats-Unis et/ou dans certains autres pays.

UNIX est une marque de The Open Group aux Etats-Unis et dans certains autres pays.

Linux est une marque de Linus Torvalds aux Etats-Unis et/ou dans certains autres pays.

Ce produit inclut des logiciels développés par le projet Eclipse (<https://www.eclipse.org/>).

Java ainsi que tous les logos et toutes les marques incluant Java sont des marques d'Oracle et/ou de ses sociétés affiliées.



Référence :

(1P) P/N: